4 INPUT/OUTPUT SUBSYSTEM

The CRAY EL series computer system input/output subsystem (IOS) acts as a preprocessor and interface between the mainframe section (central processing unit and memory) and the various customer-selected peripheral devices. The IOS 0 subassembly is required for proper system operation and must reside in the primary system cabinet. IOS 0 is defined as the IOS connected to the deadstart boot device, and is the first IOS enabled during the boot sequence. Further, IOS 0 acts as the master IOS, booting the other connected IOSs and arbitrating the access to the IOSNET. IOS 0 provides system connection to the maintenance small computer system interface (SCSI), the system console, and the maintenance workstation model EL (MWS-EL). This section describes devices incorporated within the IOS as well as their functions.

Basic Architecture

All of the IOSs used in the CRAY EL series computer system conform to the VMEbus industry standard, which specifies both electrical and mechanical rules governing the backplane. The VME backplane distributes power and delivers data address and control signals to the circuit boards plugged into the backplane. Any circuit board constructed to VMEbus standards is hardware compatible with the IOS within the CRAY EL series system.

The VME backplane used in the CRAY EL series system is manufactured to fit a standard 19-inch rack. (Refer to Figure 4-1.) It has 20 spaces, or slots, available for circuit boards, but is mechanically separated into two separate sections, each having 10 slots. Each half of the backplane contains a bus jumper board, located across slots 6 and 7. This jumper board can be removed to convert the 10-slot VME backplane into a 6-slot and a 4-slot backplane. This conversion enables the VME card cage to function as if it were as many as 4 independent IOSs in the CRAY EL series system. One VME card cage is installed in each of the cabinets of the system, making it possible to have a maximum of 16 IOSs in a 4-cabinet system.

The VME uses three types of printed circuit boards: the CPU board, memory board, and I/O board. The CPU board provides computational functions; the memory board is used as a temporary cache for data storage; and the I/O board is an interface between a peripheral device and the bus. One printed circuit board within the VMEbus is designated as the system arbitrator and always resides in slot 0 of the bus backplane.

All of the other printed circuit boards in that VMEbus are designated as slave modules. In a generic VMEbus, the physical position of the slave modules relative to the bus arbitrator is not important. However, the physical location of a slave module determines that slave's priority; the slave closest to the arbitrator has the highest priority.

The VME backplane supports the double-height 233 mm (6U) x 160 mm printed circuit boards. The full 20-slot backplane, fully loaded, is supplied by a 1000-watt power supply mounted on the back of the card cage. This power supply requires an input potential of 380 Vdc (supplied by the system bulk converter) and provides the required output voltages of –12 Vdc, +12 Vdc, and +5 Vdc.

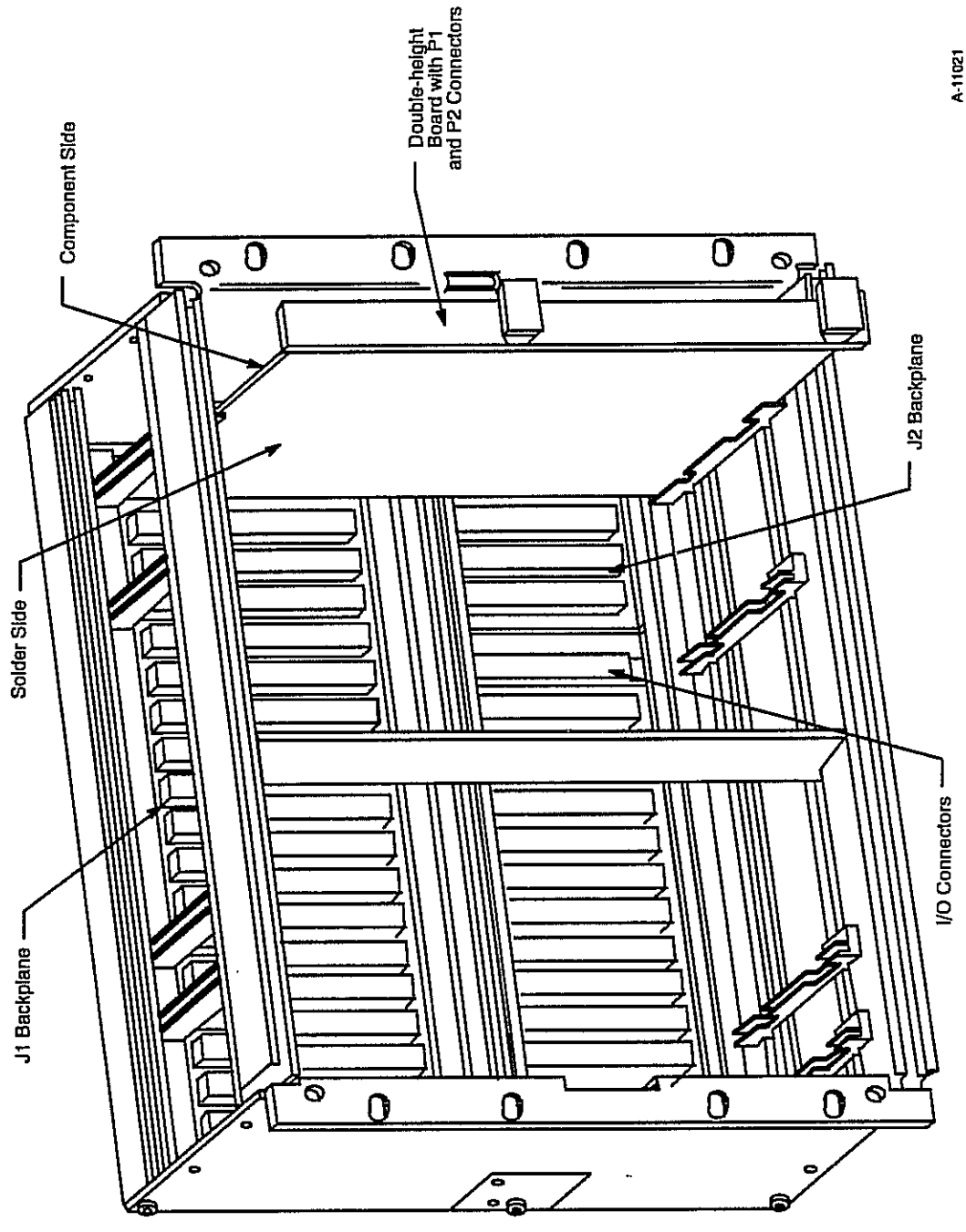


Figure 4-1. Example of VMEbus Card Cage

IOS Communications Paths

The IOS uses three communication paths. The first of these communication paths is the VMEbus. This bus is designed as a common path for carrying data, address, control, and voltage. The VMEbus is provided across connector, J1/P1. The pin assignments for this connector is shown in Table 4-1; descriptions of the signals are provided in Table 4-2. The VMEbus uses 16-, 32-, or 64-bit words; Cray Research, Inc. Entry Level Systems (CRI ELS) has elected to use the 32-bit word option. The 32-bit VME words can be transferred along the VMEbus at a peak transfer rate of 40 Mbytes/s.

The second communication path for the IOS is the Y1 bus, a channel designed by CRI ELS, to connect the IOS to the mainframe processor. This channel connects the CC ASIC in the processor to an associated input/output buffer board (IOBB) in the IOS. Each IOS has an IOBB, which is a CRI proprietary circuit board. The Y1 channel uses a 32-bit data word and supports 40-Mbyte/s peak data transfers.

The third communication path for the IOS is the IOSNET, a network that joins the various IOSs and connects them to the MWS-EL. The IOSNET uses an RS-422 port protocol with a serial data stream. This IOSNET transfers data at a rate of 1,228,800 bits per second, and is used for maintenance functions and to deadstart the IOSs in order.

Table 4-1. VME Backplane Wire List and Pin Assignment for J1/P1 Connectors

Pin Number	Row a Signal Mnemonic	Row b Signal Mnemonic	Row c Signal Mnemonic
1	D00	BBSY'	D08
2	D01	BCLR'	D09
3	D02	ACFAIL'	D10
4	D03	BG0IN'	D11
5	D04	BG0OUT'	D12
6	D05	BG1IN'	D13
7	D06	BG1OUT'	D14
8	D07	BG2IN'	D15
9	GND	BG2OUT'	GND
10	SYSCLK	BG3IN	SYSFAIL'
11	GND	BG3OUT'	BERR'
12	DS1'	BR0'	SYSRESET'
13	DS0'	BR1'	LWORD'
14	WRITE'	BR2'	AM5
15	GND	BR3'	A23
16	DTACK'	AM0	A22
17	GND	AM1	A21
18	AS'	AM2	A20
19	GND	AM3	A19
20	IACK'	GND	A18
21	IACKIN'	SERCLK(1)	A17
22	IACKOUT'	SERDAT'(1)	A16
23	AM4	GND	A15
24	A07	IRQ7'	A14
25	A06	IRQ6'	A13
26	A05	IRQ5'	A12
27	A04	IRQ4'	A11
28	A03	IRQ3'	A10
29	A02	IRQ2'	A09
30	A01	IRQ1'	A08
31	–12V	+5VSTDBY	+12V
32	+5V	+5V	+5V

NOTE: The ' designates a NOT value.

All of the other printed circuit boards in that VMEbus are desig
slave modules. In a generic VMEbus, the physical position of t
modules relative to the bus arbitrator is not important. Howeve
physical location of a slave module determines that slave's prio
slave closest to the arbitrator has the highest priority.

The VME backplane supports the double-height 233 mm (6U) x
printed circuit boards. The full 20-slot backplane, fully loaded,
supplied by a 1000-watt power supply mounted on the back of
cage. This power supply requires an input potential of 380 Vdc
(supplied by the system bulk converter) and provides the requir
voltages of –12 Vdc, +12 Vdc, and +5 Vdc.

Table 4-2. VMEbus Signal Descriptions (continued)

Signal Mnemonic	Signal Name and Description
IRQ1' – IRQ7'	*Interrupt request* ($2^0 - 2^7$) – open-collector driven signals, generated by an interrupter, which carry interrupt requests. When several lines are monitored by a single interrupt handler the highest-numbered line is given the highest priority.
LWORD'	*Longword* – a three-state driven signal used in conjunction with DS0', DS1', and A01 to select which byte locations within the 4-byte group are accessed during the data transfer.
RESERVED	*Reserved* – a signal line reserved for future VMEbus enhancements. This line *must not* be used.
SERCLK	*Serial clock* – a totem-pole driven signal that is used to synchronize data transmission on the VMEbus.
SERDAT'	*Serial data* – an open-collector driven signal that is used for VMEbus data transmission.
SYSCLK	*System clock* – a totem-pole driven signal that provides a constant 16-MHz clock signal that is independent of any other bus timing.
SYSFAIL'	*System fail* – an open-collector driven signal that indicates a failure has occurred in the system. This signal may be generated by any board on the VMEbus.
SYSRESET'	*System reset* – an open-collector driven signal which, when low, causes the system to be reset.
WRITE'	*Write* – a three-state driven signal generated by the master to indicate whether the data transfer cycle is a read or a write operation. A high level indicates a read operation; a low level indicates a write operation.
+5 V STDBY	*+5 Vdc standby* – this line supplies +5 Vdc to devices requiring battery backup.
+5 V	*+5 Vdc power* – used by system logic circuits.
+12 V	*+12 Vdc power* – used by system logic circuits.
–12 V	*–12 Vdc power* – used by system logic circuits.

NOTE: The ' designates a NOT value.

VME IOS Priority Scheme

Priority along the VMEbus is determined by the position of the module. When a slave module wants to use the VMEbus, it must send an interrupt along the bus to the arbitrator. The arbitrator will check the bus for previous requests and determine whether the bus is currently busy. If there are no requests and if the bus is inactive, the bus will be granted to the requester.

Request and grant functions are performed along specific wires across the backplane. These wires are daisy chained through each of the slave modules along the VMEbus. If an intermediate slave module were to be removed from the bus, the request/grant daisy chain would be broken and the arbitrator module would no longer recognize any of the slave modules beyond the break. Therefore, if a slave module is removed, it is important that bus jumper blocks be inserted to provide continuation of the required daisy chain connections. Jumper blocks can be inserted either on the rear or on the front of the backplane; the connections that must be jumpered are easily recognized because they are very long pins.

The CRAY EL series system includes several vendor-supplied circuit boards that reside within the IOS VMEbus structure. One of these boards acts as the system arbitration board; the others act as I/O controllers. The I/O boards allow connection of several peripherals to the VMEbus, enable several external networks to communicate with the VMEbus, and enable the VMEbus to communicate with the CRAY EL series mainframe.

Heurikon HK68/V30

The bus arbitrator is the model HK68/V30 control board manufactured by Heurikon Corporation. This processor board uses a 68030 microprocessor as the VME system processor, which controls all VME slave interrupts. Also included on the HK68/V30 board are 4 Mbytes of local memory, 64-Kbytes of programmable read-only memory (PROM), a nonvolatile random-access memory RAM (NVRAM), a SCSI controller, two serial ports, and an LED status display.

The NVRAM is programmed at the factory and is not field replaceable. The PROM contains a very limited set of functions designed to allow the HK68/V30 board to make calls to the system boot devices and provide limited power-on diagnostics. The functions contained on the PROM are console I/O, maintenance SCSI disk I/O, software installation, IOS dumps, and system boot. The PROM is not capable of operating the system peripherals or UNICOS. These functions are subject to upgrade when the IOS firmware is upgraded by an IOS release. When the system is under PROM control, the prompt on the system console screen is `BOOT>`. After execution of the basic IOS kernel load operation, this prompt changes to `IOS>` indicating that the IOS kernel has been copied from the maintenance hard disk into the HK68/V30 local memory.

SCSI Controller

The onboard SCSI controller section provides a SCSI bus connection to the maintenance peripherals, which include a 204-Mbyte hard disk drive provided by Seagate Technology, Inc. and a 1.3-Gbyte quarter-inch cartridge (QIC) tape drive. The peripherals may also include an optional EXABYTE Corporation 5-Gbyte 8-mm helical scan tape drive. The maintenance SCSI and its peripherals are intended for use by CRI maintenance personnel, but the customer may use the tape drives to make system backup tapes when no other backup method is available.

Serial Ports

Both serial ports are operated by a Zilog Z8530 SCC integrated circuit. One of the serial ports is operated at 19,200 baud in accordance with the RS-232 protocol. This port is connected externally to the system console, a WYSE 60 terminal. This connection enables the customer to directly access the master IOS for system administration. The second serial port operates as a high-speed RS-422 port with a 1,228,800 bit-per-second connection. This RS-422 port, called IOSNET, connects to the MWS-EL and to all other IOSs in the system. The IOSNET is used for technical maintenance connection and as the IOS deadstart boot path. Individual IOS status is reported using this bus. For proper system operation, the RS-422 port must either be connected to another device or terminated (the terminator part number is 90156500) at the bulkhead.

Input/Output Buffer Board

The second printed circuit board within the IOS is the input/output buffer board (IOBB), designed and manufactured by CRI ELS. The IOBB provides up to 16 Mbytes of onboard buffer memory for the IOS and acts as a communications interface between the VMEbus devices and the mainframe processor. The block diagram of the IOBB is shown in Figure 4-1. Note that the IOBB provides both VMEbus and Y1 bus logic and also provides a 4-bit parity protection circuit. The parity scheme used on the IOBB is odd parity.

Communication between the IOS and mainframe processor takes place in the following manner: the IOP generates an interrupt to initiate a mainframe processor request, and the processor generates interrupts to initiate a peripheral read or write operation or to terminate the processor request. These interrupts generate I/O task control blocks (IOTCBs), of which there are two types: I/O IOTCB and CONSOLE IOTCB. Both of these types have the same format with minor variations, as described in the following subsections.

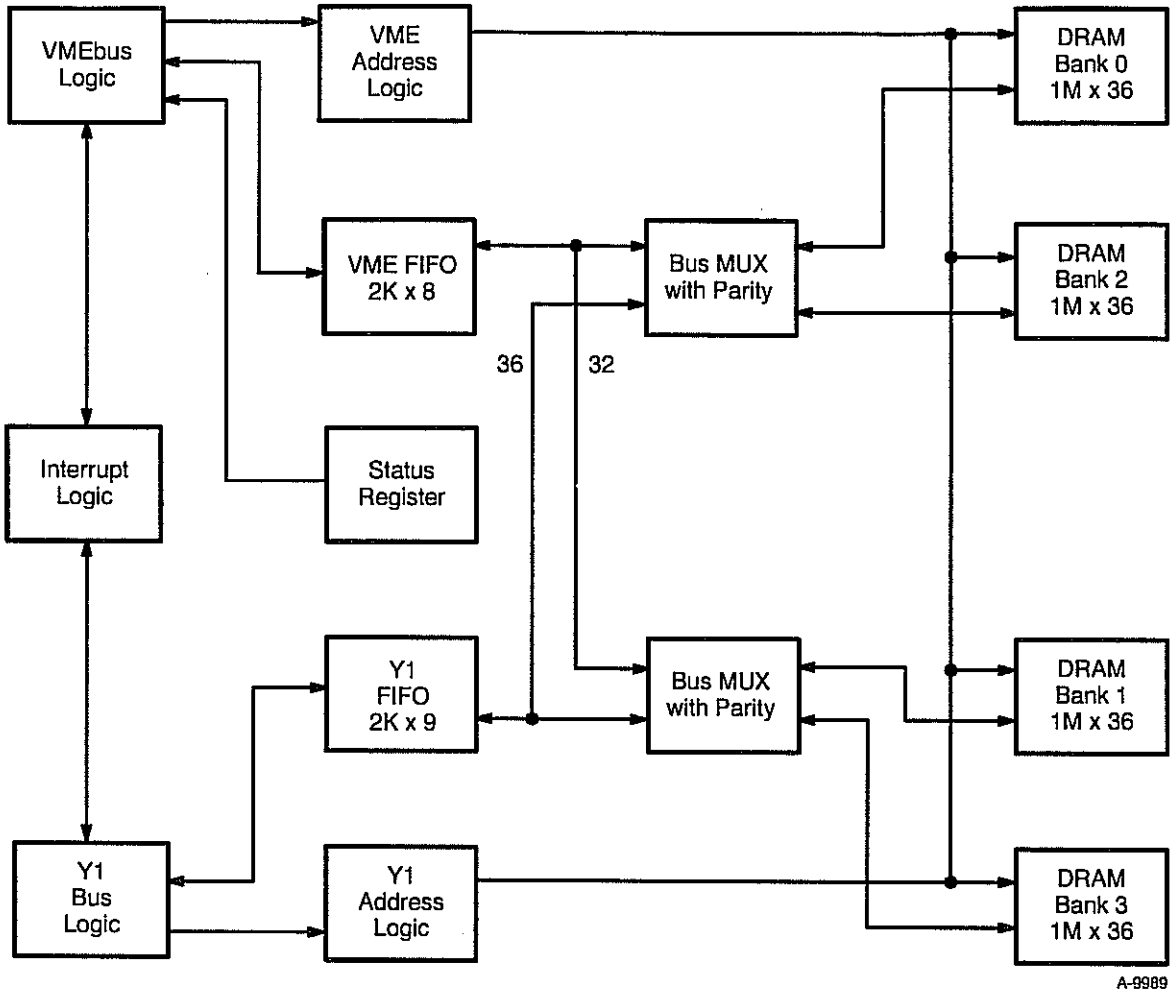


Figure 4-1. IOBB Block Diagram

I/O IOTCB Format

When the CRAY EL series system is booted, an IOTCB pool is created in main memory. This IOTCB pool contains 5,000 four-word IOTCBs. An IOTCB word is 32 bits long, which conforms to VMEbus and Y-1 bus protocol. It is possible to have all 5,000 IOTCBs formatted and waiting for execution, but the HK68/V30 can queue only 7 active IOTCBs. The IOTCBs are accessed by using pointers that indicate the current IOTCB, the next IOTCB, and the final IOTCB.

The format of an I/O IOTCB is shown in Figure 4-3, and an explanation of the individual fields follows the figure.

Once an I/O IOTCB begins to operate, the IOP can move to other IOS functions. These other functions continue until the IOP receives an interrupt indicating completion from the I/O IOTCB. Then the IOP returns to the I/O IOTCB to check status and look for parity errors. If no errors are reported, the IOP can move on to the next I/O IOTCB in the queue or to another task.

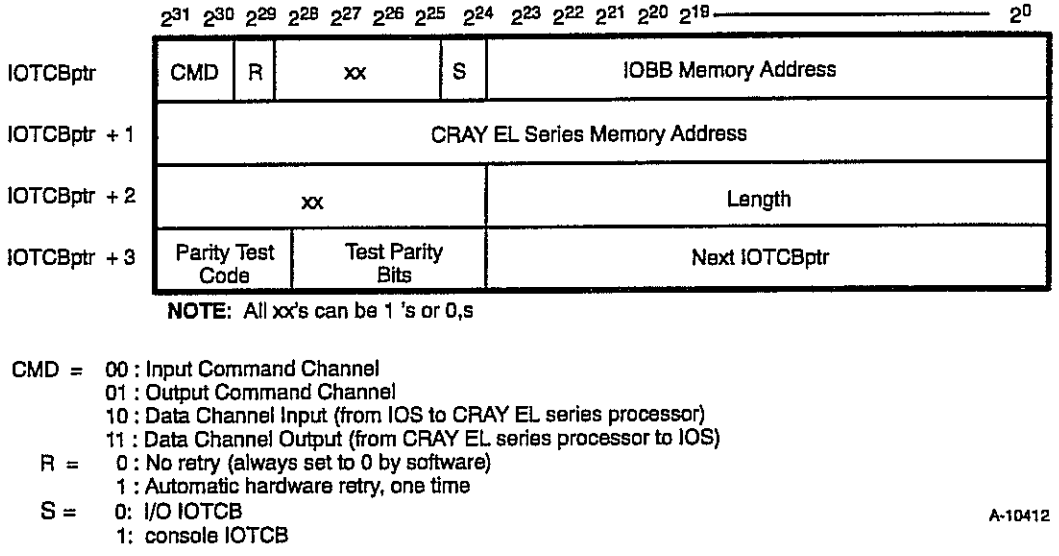


Figure 4-3. I/O IOTCB Format

IOBB Memory Address

IOBB memory address is the starting address in IOBB memory that data should be read from or written to. Twenty-two bits can address up to 4 million 32-bit words (16 Mbytes) of IOBB memory. The IOBB memory address is a 32-bit word address.

CRAY EL Series Memory Address

CRAY EL series memory address is the starting address that data should be read from or written to. This field is ignored by the CC ASIC if the command = 0x.

Length

Length is the number of 32-bit words to be transferred, limited by the total amount of memory on the IOBB. The length must be even (for example, CC ignores bit 0). This field is ignored by the CC if CMD = 0x.

NOTE: Length = 0 means a no-operation condition (data channel only); however, IOTCBptr is loaded and completion interrupt is generated normally.

Next IOTCBptr

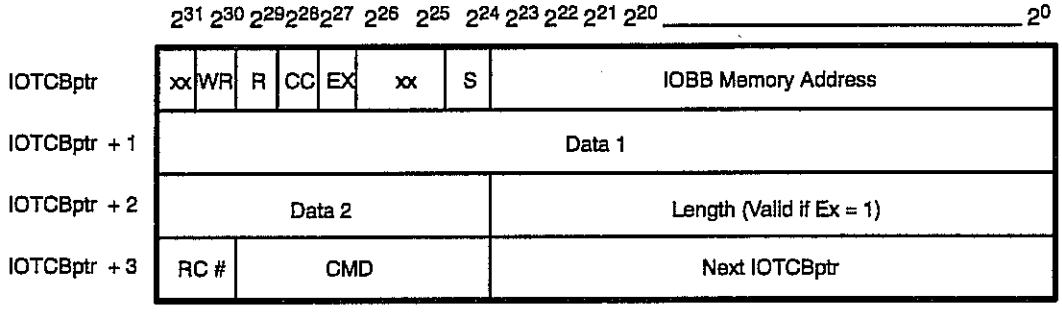
The IOBB memory address where the next IOTCB resides is the next IOTCBptr. The next IOTCBptr is a 32-bit word address.

CONSOLE IOTCB Format

The format of a CONSOLE IOTCB is shown in Figure 4-4.

When a CONSOLE IOTCB has started, the IOS waits for it to complete before moving on to another task.

Once a channel interrupt has been received, the IOP initiates an IOTCB, sends an IOTCB pending interrupt to the CC ASIC on the processor, and waits for the processor to acknowledge the interrupt. When the interrupt is received by the processor, it is handled by the CC ASIC, which acts as the IOTCB processor within the processor. The operating sequence for the channel operation is listed in the following subsections.



RC# = 00 : Command is for RC chip of processor number 0
01 : Command is for RC chip of processor number 1
10 : Command is for RC chip of processor number 2
11 : Command is for RC chip of processor number 3

CC = 1 : Local operation (local to CC), no need to initiate any CONBUS cycles

EX = 1 : Extra data; the following fields are valid:
IOBB memory address – starting address
length – number of 32-bit words transferred

WR = 0 – Write from IOBB
WR = 1 – Write to IOBB

R = 0 : No retry (same as I/O IOTCB)
1 : automatic hardware retry, one time

S = 0 : I/O IOTCB
1 : Console IOTCB

A-10413

Figure 4-4. CONSOLE IOTCB Format

IOS Error Types and Explanations

The CRAY EL series system IOS experiences three typed of software errors that are all fatal and may cause the system to crash. IOS panics, UNICOS panics, and IOS exception faults are described in the following subsections.

IOS Panics

The type of fatal IOS error most likely to occur is the IOS panic message, which is caused by extensive use of an assertion panic macro throughout the code. This macro is designed to detect fatal conditions caused by inconsistent states of some cooperative routines, catastrophic hardware failures, resource exhaustion, or timer expirations.

The panic message is a standard format displayed both on the IOS console and entered into `/adm/syslog`. A typical panic message appears on the screen as:

```
IOS ASSERTION PANIC:  err
MSG:  cc_init:  qcreate  QI  XP,  err= c
file:  cc_init.c,  Line:  144,   Date:  24/04/91
15:14:05
Sccsid:  @(#)cc_init.c  7.1  cc_init.c  7.1
91/02/27  09:59:17
Version:  Entry Level Systems IOS version 7.3
91/03/25
11:26:56

panic information saved in /adm/syslog—
```

The first statement after the IOS assertion panic line contains the key information for analyzing the problem. The remaining information enables ELS software developers to verify that the IOS version is a supported version, and provides a line number in the source code where the panic took place. Panics must be analyzed by first categorizing the primary message into one of several types. They are VIOSX error codes, state errors, range or bound errors, and fatal hardware status errors. Panic messages can be classified as follows:

- VIOSX errors: messages that contain the words *pend, post, qpend, qpost, rblock, gblock,* or *task.*
- State errors: messages that contain the word *state* with an expected and actual value.

- Range errors: messages that present a numeric value of a variable with actual and expected values. For example, the first three lines of a panic message may be:

```
IOS ASSERTION PANIC:  (((long)112 &
0xfff00000) !=(IOBB_MIN_ADDR)
MSG:  invalid IOTCB ptr 0x0
```

- Fatal hardware errors: messages that show an expected and actual status from a named device.

The VIOSX assertion panic macros must be analyzed further, using the VIOSX/32 users guide to determine what to do next. The guide has a description of all the VIOSX calls, which are pend, post, qpend, qpost, rblock, gblock, tsuspend, tdelete, etc. To determine the nature of the error, look up the call in the book; for each call there is a short list of possible error codes.

UNICOS Panics

The IOS has a specific message for displaying UNICOS panic messages. Here is an example:

```
UNICOS SYSTEM PANIC
```

If the console was under UNICOS control, type **CNTL-a** to switch to IOS control and review the error message. Here is an example:

```
IOS NOTICE: PANIC:  trap.s:  jump to absolute
zero encountered
File:  cc_task.c,  Line:  304,  Date:  24/04/91
15:19:33
syncing disk . . .
Rebooting IOS . . .
```

IOS Exception Faults

These types of errors are high-severity faults that produce an IOS processor register dump display. The exception faults are catastrophic conditions under which IOS processor instruction execution must stop. For example:

```
Exception:  Bus Error
Exception Vector Offset:  $8
Fault Address 09999999
SR = 3000  Trace = OFF Active State = MSP
Interrupt mast = 0
```

```
  PC = 00027432   D0 = 00000002   A0 = 09000000
VBR  = 03000000
USP = 00000000  D1 = FFFF0007   A1 = 00000000
SFC = 00000007
MSP = 033B7DFA  D2 = 09000000   A2 = 00027574
DFC = 00000007
ISP  = 033BD3BE   D3 = 00000002   A3 = 033B6DD8
CAAR = 1C000E20
  D4 = 00000004   A4 = 0300045A   CACR = 00000001
  D5 = 00000000   A5 = 033B6DD8
  D6 = 033BC3B8   A6 = 033B7EB6
  D7 = 00000000   A7 = 033B7DFA
00027432  2010          MOVE.L        (A0)   ,D0
IOS>
```

These error messages are essentially a dump of the IOS processor's operating registers and state information. Currently this message is not entered into the syslog because the fault causes the system to crash.

System Dumps

Two types of dumps may be saved for analysis of system crashes and problems: IOS dumps and UNICOS dumps.

IOS Dumps

The command `iosdump` saves both the IOS processor's memory and IOBB memory to the SCSI hard disk. This command is invoked from the `IOS>` prompt. An IOS dump is automatically generated if an IOS panic occurs.

The format of the command is:

```
iosdump filename
```

EXAMPLE: `IOS>iosdump /adm/iosdump1`

Remember that you can generate an IOS dump from PROM software in the IOS. Therefore, if the IOS appears to hang, you can press the reset button on the IOS to load the basic kernel from PROM. The RAM contents and IOBB RAM contents will remain intact, so you can generate a useful dump of the IOS.

UNICOS Dumps

The command `mfdump` captures mainframe memory in a file on a system disk beginning at memory address 0 and continuing for the number of words specified in the word count field (default is 16 Mwords.) This command is invoked from the IOS> prompt. A dump should be generated whenever a UNICOS system panic or an IOS panic occurs. The partition that receives the dump must be pre-initialized with the `idmp` command. The `mfdump` command stores its runtime parameters in a binary file. These parameters may be viewed using the `-c` command line option and changed using the `-q` command line option. Dump files are then processed with the `cpdmp` command.

The initialization (`idmp`) and processing (`cpdmp`) are usually performed in the multiuser startup script `/etc/coredd`. After `cpdmp` processing, dump files may be analyzed with the crash utility.

NOTE: The `mfdump` command halts a running system.

The format of the command is as follows:

```
mfdump [-c f q r v]
```

Command Input Channel

The command input channel controls data transfers from the IOBB to the CPU. Five registers in the CC ASIC are associated with each input channel as follows:

- Channel address (CA) -- the starting memory address in CRAY EL series memory
- Channel limit (CL) -- the ending memory address in CRAY EL series memory
- Channel error (CE) flag
- Channel interrupt (CI) flag
- Channel number (C#)

The operating sequence of transferring data from the IOBB to the CC is as follows:

1. The processor loads the CL register.
2. The processor loads the CA register; the corresponding input channel (C#) is opened.

3. The CC sends a Ready to Receive Return Status Block (RSB) interrupt signal to the MIOP via the IOBB.

4. The MIOP sets up an RSB in IOBB memory.

5. The MIOP sets up an IOTCB (CMD = 00, IOBB memory address = starting address of RSB to be read).

6. The MIOP sends an IOTCB Pending interrupt signal to the CC via the IOBB.

7. The CC fetches IOTCB (pointed to by IOTCBptr), picks up the IOBB memory address, uses the CA and CL registers instead of the CRAY EL series memory address from IOTCB, and completes the transfer.

8. The CC interrupts the CRAY EL series processor when CA = CL.

9. The CC sends an IOTCB Done interrupt signal to MIOP via IOBB.

Command Output Channel

The command output channel handles data transfers from the CC ASIC, in the selected processor in the mainframe, to the selected IOBB in the IOS. Five registers in the CC are associated with each output channel as follows:

- Channel address (CA) -- the starting memory address in CRAY EL series memory
- Channel limit (CL) -- the ending memory address in CRAY EL series memory
- Channel error (CE) flag
- Channel interrupt (CI) flag
- Channel number (C#)

The operating sequence of an output channel is as follows:

1. The processor sets up a control block (CB) in CRAY EL series memory.

2. The processor loads the CL register.

3. The processor loads the CA register; the corresponding output channel (C#) is opened.

4. The CC sends a CB Pending interrupt signal to the MIOP via the IOBB.

5. The MIOP sets up an IOTCB (CMD = 01, IOBB memory address = starting address for loading CB).

6. The MIOP sends an IOTCB Pending interrupt signal to the CC via the IOBB.

7. The CC fetches the IOTCB (pointed to by the IOTCBptr register), picks up the IOBB memory address, uses the CA and CL registers instead of the CRAY EL series memory address from IOTCB, and completes the transfer.

8. The CC interrupts the processor when CA = CL.

9. The CC sends an IOTCB Done interrupt signal to the MIOP via the IOBB.

Data Channel (Input and Output)

The operating sequence of a data channel is as follows:

1. The MIOP sets up an IOTCB (CMD = 10 or 11).

2. The MIOP sends an IOTCB Pending interrupt signal to the CC via the IOBB.

3. The CC fetches the IOTCB (pointed to by the IOTCBptr register), interprets all parameters from the IOTCB, and completes the transfer.

4. The CC sends an IOTCB Done interrupt signal to the MIOP via the IOBB.

The IOBB also detects and reports errors. The Y1 bus enables parity transfers between the processor in the mainframe and the IOBB. The IOBB can thus detect and report parity errors, of which there are two types: errors during IOTCB fetch and errors during IOTCB execution. If a parity error occurs during the IOTCB fetch operation, the CC ASIC is not allowed to complete the IOTCB. The CC clears its IOTCB pending queue, resets the IOTCBptr register to 0, and sends an IOTCB fetch error interrupt to the MIOP via the IOBB. To restart, the MIOP must load the next IOTCB to memory location 0 on the IOBB, which is the default beginning of the IOTCB chain.

Errors that occur during IOTCB execution can also be of two types: command channel errors or data channel errors. In both cases, the execution of the IOTCB continues to completion; then the CC ASIC initiates a retry if the retry bit is set in the IOTCB. If this retry is not successful, an error message (IOTCB execution error) is returned to the MIOP. At this point, the operating system determines the procedure to follow for further error resolution.

Communications

The communications channel in the CRAY EL series system is the Y1 bus. The Y1 bus is capable of 40-Mbyte/s transfer rates. The actual data signal is a differentiated value that provides a high level of noise immunity.

Figure 4-5 shows how the IOBB fits into the communication path in the CRAY EL series system. Note that the IOBB is a slave device to the IOP connected to it via the VMEbus.

Each of the Y1 buses has a channel pair number. Because the Y1 bus is bidirectional, it can be considered a channel pair. Channel pair assignments are as follows:

- Processor 0
  - CC0 - channels 20/21
  - CC0 - channels 22/23
  - CC1 - channels 24/25
  - CC1 - channels 26/27
- Processor 1
  - CC0 - channels 40/41
  - CC0 - channels 42/43
  - CC1 - channels 44/45
  - CC1 - channels 46/47
- Processor 2
  - CC0 - channels 60/61
  - CC0 - channels 62/63
  - CC1 - channels 64/65
  - CC1 - channels 66/67
- Processor 3
  - CC0 - channels 100/101
  - CC0 - channels 102/103
  - CC1 - channels 104/105
  - CC1 - channels 106/107
- Processor 4

In this assignment scheme, the even-numbered channels are input channels, and the odd-numbered channels are output channels. The Y1 bus is bidirectional in the CRAY EL series system; therefore the association of channel numbers is for reference purposes only.

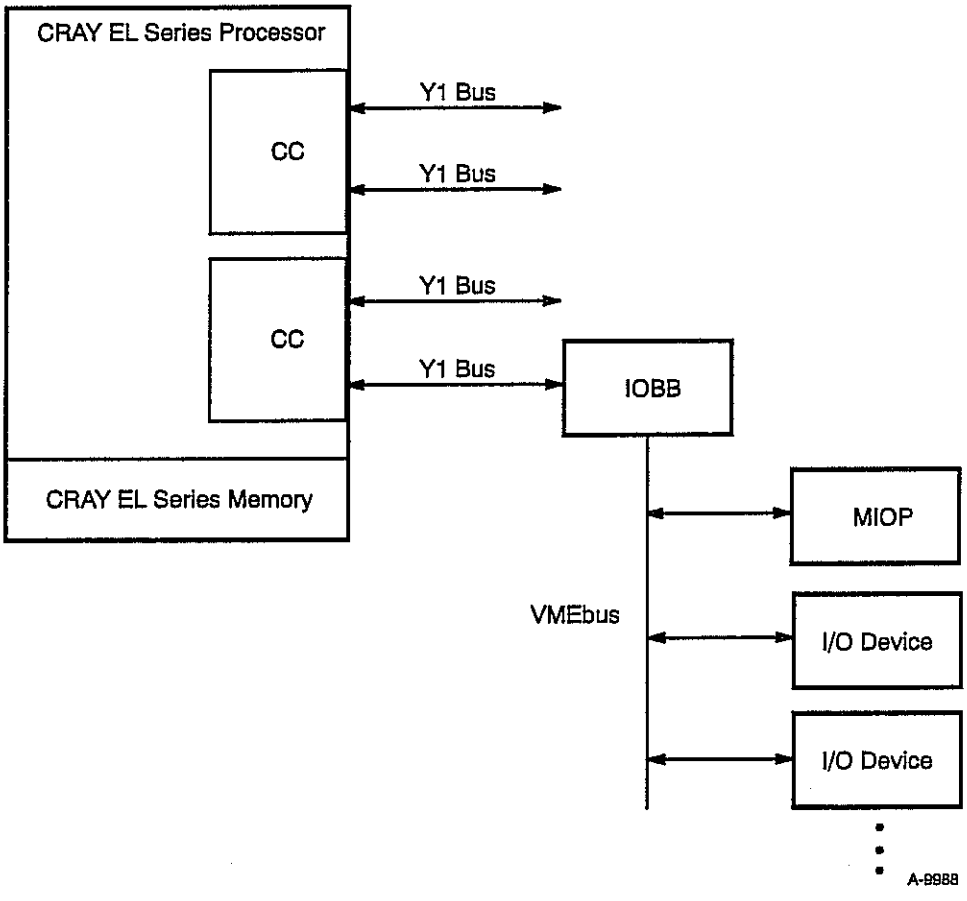


Figure 4-5. IOBB Communication

Y1 Bus Interface

The Y1 bus is a Cray Research, Inc. proprietary bus that connects the IOBB, located in the IOS, to the CC located in the processor. The Y1 bus transfers data at high speeds over long distances between the IOBB and the CC. The Y1 bus has the following characteristics:

- Point-to-point connection from the IOBB to memory; the IOBB is a slave memory

- The data bus is bidirectional, is 32 bits wide, and has byte parity
- Address, control, and data are time-multiplexed using the single 32-bit data bus
- Maximum cable length is 50 ft
- Capable of passing interrupts in both directions at any time (independent of data transfers)
- All signals are differential for high noise immunity

Y1 Bus Signal Descriptions

Figure 4-6 shows the Y1 bus signals. The IOBB is in receive mode until conditioned by the CC bus signal. The subsections that follow describe each signal.

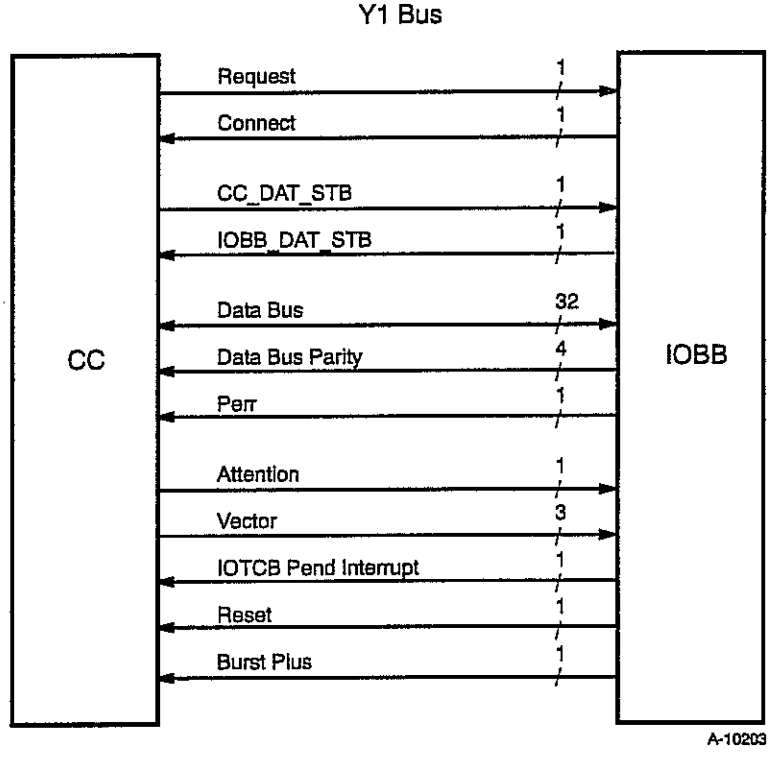


Figure 4-6. Y1 Bus Signal Descriptions

Request Signal

The CC communicates with the IOBB by placing the following information on the Data Bus signal. Refer to Table 4-3.

Table 4-3. Data Bus Information

Bits	Definition
$2^{21} - 2^{0}$	IOBB memory address (addresses 4M 32-bit words)
$2^{28} - 2^{22}$	Reserved (driven low)
$2^{30} - 2^{29}$	00 = 32-word data burst
$2^{30} - 2^{29}$	01 = 64-word data burst
$2^{30} - 2^{29}$	10 = 128-word data burst
$2^{30} - 2^{29}$	11 = 4-word burst
2^{31}	1 = Data transfer from CC to IOBB (write)
2^{31}	0 = Data transfer from IOBB to CC (read)

The CC then asserts the Request signal after this information has been valid on the Data Bus signal for a minimum of 50 ns. This information should remain valid on the data bus until the IOBB asserts the Connect signal. The IOBB can therefore use the rising edge of the Request signal to latch the information. The Request signal can be reasserted only when the Connect signal has been negated. Each Request and Connect signal sequence can transfer a burst of 4, 32, 64, or 128 thirty-two-bit words of data.

Connect Signal

The IOBB responds to a Request signal by asserting the Connect signal to indicate that the address, direction, and burst size are latched. The CC should keep the address and direction on the bus until the Connect signal is asserted by the IOBB. The Request and Connect signals should remain asserted for the duration of the transfer, except during read operations in which the CC may drop the Request signal after receiving a Connect signal from the IOBB. At the end of a write operation, the CC should negate the Request signal. The IOBB responds by negating the Connect signal when it also detects the end of the transfer.

CC_DAT_STB

The CC_DAT_STB signal is used to send data from the CC to the IOBB. After connection is established by the Request and Connect signals (and the transfer is from CC to IOBB), 4, 32, 64, or 128 CC_DAT_STB pulses (bursts of data) should be sent to the IOBB. The maximum frequency of the CC_DAT_STB signal is one pulse every 50 ns, so the maximum burst transfer rate is 80 Mbytes/s. If the data transfer ends at a nonburst boundary, the last burst is less than 4, 32, 64, or 128 words.

Burst Plus Signal

When the Burst Plus signal is asserted, it informs the CC that the new IOBB is present so that faster burst-transfer rates are possible and so that burst lengths of 4, 32, 64, and 128 words are supported.

IOBB_DAT_STB Signal

The IOBB_DAT_STB signal is used to send data from the IOBB to the CC. After connection is established by the Request and Connect signals (and the transfer is from IOBB to CC), 4, 32, 64, or 128 IOBB_DAT_STB pulses should be sent to the CC. The maximum frequency of the IOBB_DAT_STB signal is one pulse every 50 ns, so the maximum burst-transfer rate is 80 Mbytes/s.

Data Bus and Data Bus Parity Signals

The Data Bus signal is a 32-bit bidirectional data bus with odd byte parity. The P3 signal (parity, byte 3) is for bits 2^{31} through 2^{24}; the P2 signal (parity, byte 2) is for bits 2^{23} through 2^{16}; the P1 signal (parity, byte 1) is for bits 2^{15} through 2^{8}; and the P0 signal (parity, byte 0) is for bits 2^{7} through 2^{0}. Data, address, direction, and burst length information is transferred via the Data Bus signal.

Perr Signal

The IOBB uses the Perr signal to report a parity error of address or data to the CC. The IOBB sends the Perr signal pulse whenever it detects a parity error during the data or address transfer sequence. The CC does not terminate the transfer when a parity error is detected. When an address parity error is detected in the IOBB, the CC continues with the sequence of data transfers, but the IOBB does not write the data into its memory.

Attention and Vector Signals

The Attention signal is the interrupt signal from the CC to the IOBB. The 3 Vector signal bits indicate the reason for only the interrupt. Refer to Table 4-4. These signal bits can be observed by using an oscilloscope and are not assessed during normal field maintenance.

Table 4-4. Attention and Vector Information

Vector 2^2	2^1	2^0	Cause for the Interrupt
1	1	1	Console interrupt
1	1	0	CA is loaded to the input channel (ready to receive RSB)
1	0	1	CA is loaded to the output channel (CB pending)
1	0	0	Reset opcode is received in the output channel
0	1	1	IOTCB done
0	1	0	IOTCB fetch error
0	0	1	IOTCB execution error
0	0	0	Not used

The CC may send consecutive interrupts to the IOBB at a rate of one pulse (50 ns) every 330 ns. The IOBB saves the interrupts and vectors in the status register for the IOP to reference.

IOTCB Pend Interrupt Signal

The I/O task control block (IOTCB) is used by the IOP to request a data transfer between CRAY EL series system memory and the IOBB. When an IOTCB is ready in the IOBB memory, the IOBB interrupts the CC. The interrupt is a pulse 50 ns (minimum) wide. The CC can queue as many as seven IOTCB Pend Interrupt signal pulses without responding to them.

Reset Signal

The Reset signal resets the CRAY EL series channel to which the IOP is connected. This signal can be used to force the CC to search for the next IOTCB at location 0 in the IOBB memory.

Y1 Bus Pin Assignments

Two 60-pin connectors, P1 and P2, provide an interface between the IOBB and the Y1 bus. These connectors are located on the front panel of the board. Refer to Table 4-5 and Table 4-6.

Table 4-5. P1 Pin Assignments

P1 Pin	Signal Name	P1 Pin	Signal Name	P1 Pin	Signal Name
1	BURSTPLUS	21	GND	41	DB15P
2	GND	22	GND	42	DB15N
3	GND	23	DB08P	43	DB16P
4	GND	24	DB08N	44	DB16N
5	DB00P	25	DB09P	45	DB17P
6	DB00N	26	DB09N	46	DB17N
7	DB01P	27	DB10P	47	DB18P
8	DB01N	28	DB10N	48	DB18N
9	DB02P	29	DB11P	49	DB19P
10	DB02N	30	DB11N	50	DB19N
11	DB03P	31	Not Used	51	DB20P
12	DB03N	32	Not Used	52	DB20N
13	DB04P	33	DB12P	53	DB21P
14	DB04N	34	DB12N	54	DB21N
15	DB05P	35	DB13P	55	DB22P
16	DB05N	36	DB13N	56	DB22N
17	DB06P	37	DB14P	57	GND
18	DB06N	38	DB14N	58	GND
19	DB07P	39	GND	59	GND
20	DB07N	40	GND	60	GND

Table 4-6. P2 Pin Assignments

P2 Pin	Signal Name	P2 Pin	Signal Name	P2 Pin	Signal Name
1	GND	21	GND	41	REQP
2	GND	22	GND	42	REQN
3	GND	23	DB31P	43	CONNP
4	GND	24	DB31N	44	CONNN
5	DB23P	25	DBP0P	45	TCBPENDP
6	DB23N	26	DBP0N	46	TCBPENDN
7	DB24P	27	DBP1P	47	PERRP
8	DB24N	28	DBP1N	48	PERRN
9	DB25P	29	DBP2P	49	VECT2P
10	DB25N	30	DBP2N	50	VECT2N
11	DB26P	31	RESET	51	VECT1P
12	DB26N	32	GND	52	VECT1N
13	DB27P	33	DBP3P	53	VECT0P
14	DB27N	34	DBP3N	54	VECT0N
15	DB28P	35	IOCCDRDYP	55	ATTNP
16	DB28N	36	IOCCDRDYN	56	ATTNN
17	DB29P	37	IOBBDRDYP	57	GND
18	DB29N	38	IOBBDRDYN	58	GND
19	DB30P	39	GND	59	GND
20	DB30N	40	GND	60	GND

Y1 Bus Channel Operations

The CRAY EL series system has two types of logical channels: command channels and data channels. Processor instructions control command input and command output channels. The processor uses the output channels to send commands to the IOS and uses the input channels to receive status from the IOS. Data channels are controlled by the IOS. The IOS uses data channels to transfer data between CRAY EL series memory and IOBB memory.

All of the channel operations are initiated by the IOP in the form of IOTCBs. The IOP has to set up an IOTCB in IOBB memory for every transfer between CRAY EL series memory and the IOBB (a maximum of 7 outstanding IOTCBs is allowed). There are two types of IOTCBs: an I/O IOTCB (S=0), and a CONSOLE IOTCB (S=1).

IOSNET

Communication between the IOP, the other IOSs, and the maintenance workstation (MWS) takes place via the IOSNET, an RS-422 serial interface that can sustain transfers of 1.2 Mbits/s. The IOSNET is not a true serial interface, with the exception of MIOP in IOS 0. If the MIOP is defective, the entire system is down. However, the remaining IOSs connect to the IOSNET in a T format, so that if one of these IOSs becomes defective, the rest of the system remains operational.

Communications along the IOSNET are arbitrated by the MIOP. If IOP 2 needs to communicate with IOP 1, the communications path includes MIOP, IOP 1, IOSNET, and IOP 2. Refer to Figure 4-7 for an illustration of this daisy chain. Figure 4-7 also shows the communication path from the MIOP to the required system peripherals via the SCSI interface. The required system peripherals are the 0.25-inch cartridge tape drive and the Winchester hard disk drive. The helical-scan tape drive is an option. The system console used with the CRAY EL series system is a WYSE model 60 that uses an RS-232 interface to the MIOP.

Note that a terminator plug is required to ensure proper network operation.

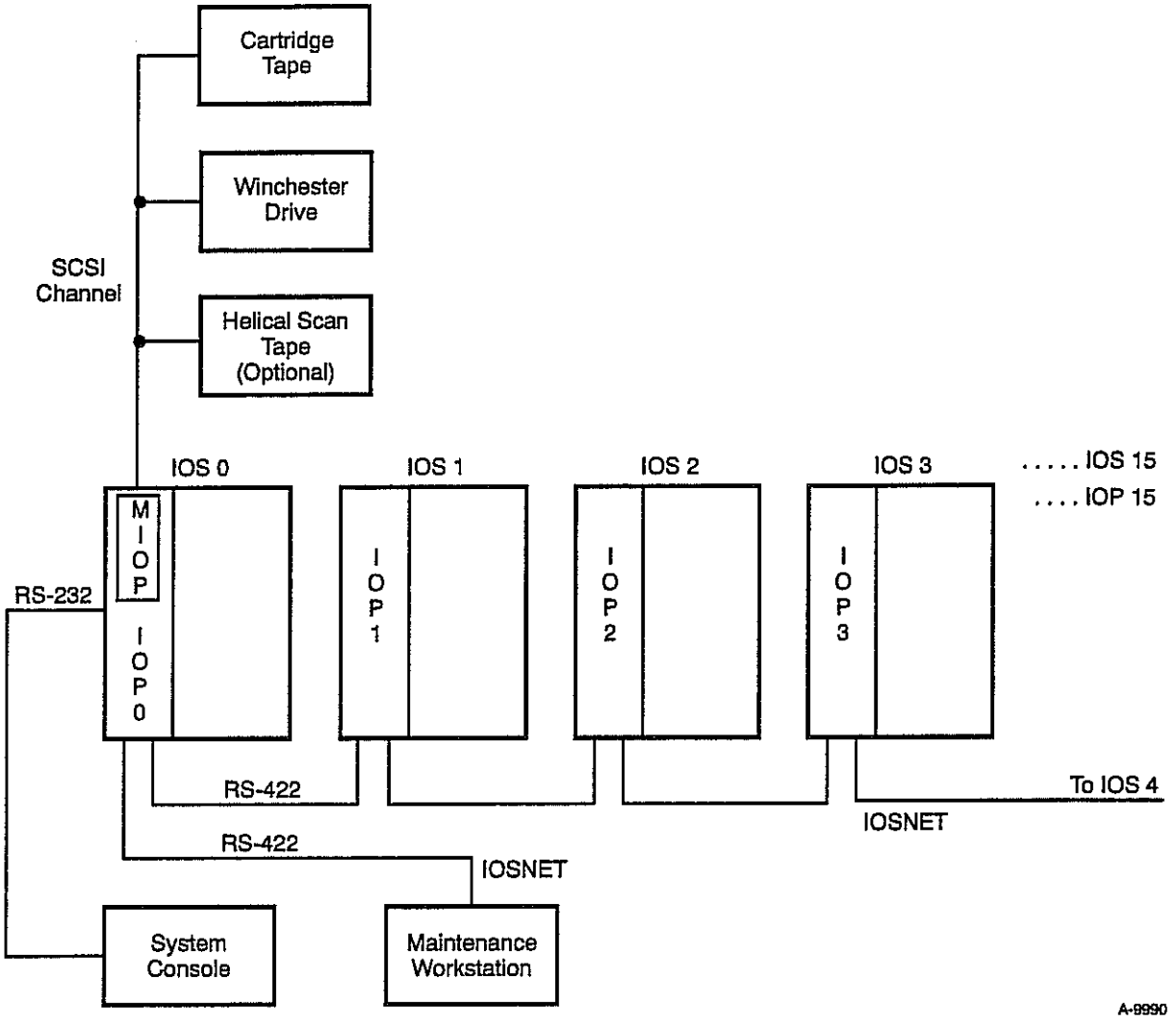


Figure 4-7. IOP Daisy Chain Configuration

IOS-supported Peripheral Controllers

The design of the IOS allows many different controllers to be installed. Consequently, many different peripheral devices can be connected to the CRAY EL series system. The minimum configuration that is available for the IOS includes a Heurikon HK68/V30 as the IOP, an IOBB, and one peripheral controller. Refer to Table 4-7 for the types of controllers supported by Cray Research, Inc.

Table 4-7. IOS-supported Peripheral Controllers

Device Type	Interface Protocol	Controller Vendor	Model Number	Peripheral Type	Cray Research Name
Tapes	SCSI	Heurikon Corporation †	HK68/V30	QIC 0.25 in.	DS-2
	SCSI	Heurikon Corporation †	HK68/V30	8-mm cartridge	EX-2
	Pertec	CIPRICO Inc.	TM3000	9-track round tape	TC-2
	SCSI	CIPRICO Inc.	RF3564	18-track cartridge (or) 8-mm cartridge (or) 18-track silo ‡	SI-1
Disks	SCSI	Heurikon Corporation †	HK68/V30	3.5 in. Winchester	HD-1
	ESDI	CIPRICO Inc.	RF3411	5.25 in. high density	DC-3
	IPI	Xylogics	SV7800	8 in. Sabre VII	DC-4
	ESDI array subsystem	Maximum Strategy	Bus master to a Strategy 2	5.25 in. ESDI array (10 disks per array)	DAS-2
	SCSI-2	Interphase Corporation	4220 Cougar	System Disk Drives (DD5-S)	SI-2
Network	Ethernet	CMC Corporation	130	Ethernet interface	EI-1
	Fiber Distributed Device Interface (FDDI)	Interphase Corporation	4211 Peregrine	Fiber-optic channel	FI-1
	FDDI	Interphase Corporation	5211 Peregrine II	Fiber-optic channel	FI-2
	HYPERchannel	NSC Corporation	PI-492	HYPERchannel	HC-1

† The drive connected to the Heurikon SCSI channel in IOS 0 is a dedicated system device. The tapes on this channel are standard equipment and can be used by the customer.

‡ A customer may connect the CIPRICO SCSI controller to a Storage Technology Corporation 4781/4780 18-track cartridge silo transport. At present, these tape devices are not maintained by Cray Research.

Characteristics of each supported controller are listed in the following subsections. Diagrams of the jumper locations, cabling, board layouts, switch settings, and block diagrams follow the subsections.

Heurikon Corporation HK68/V30 VME IOP

The pertinent information about the Heurikon board is as follows:

- Uses 68030 microprocessor
- Acts as VMEbus arbitration control
- Acts as SCSI bus initiator
- The *Heurikon HK68/V30 User's Manual* describes the status LEDs and user LEDs

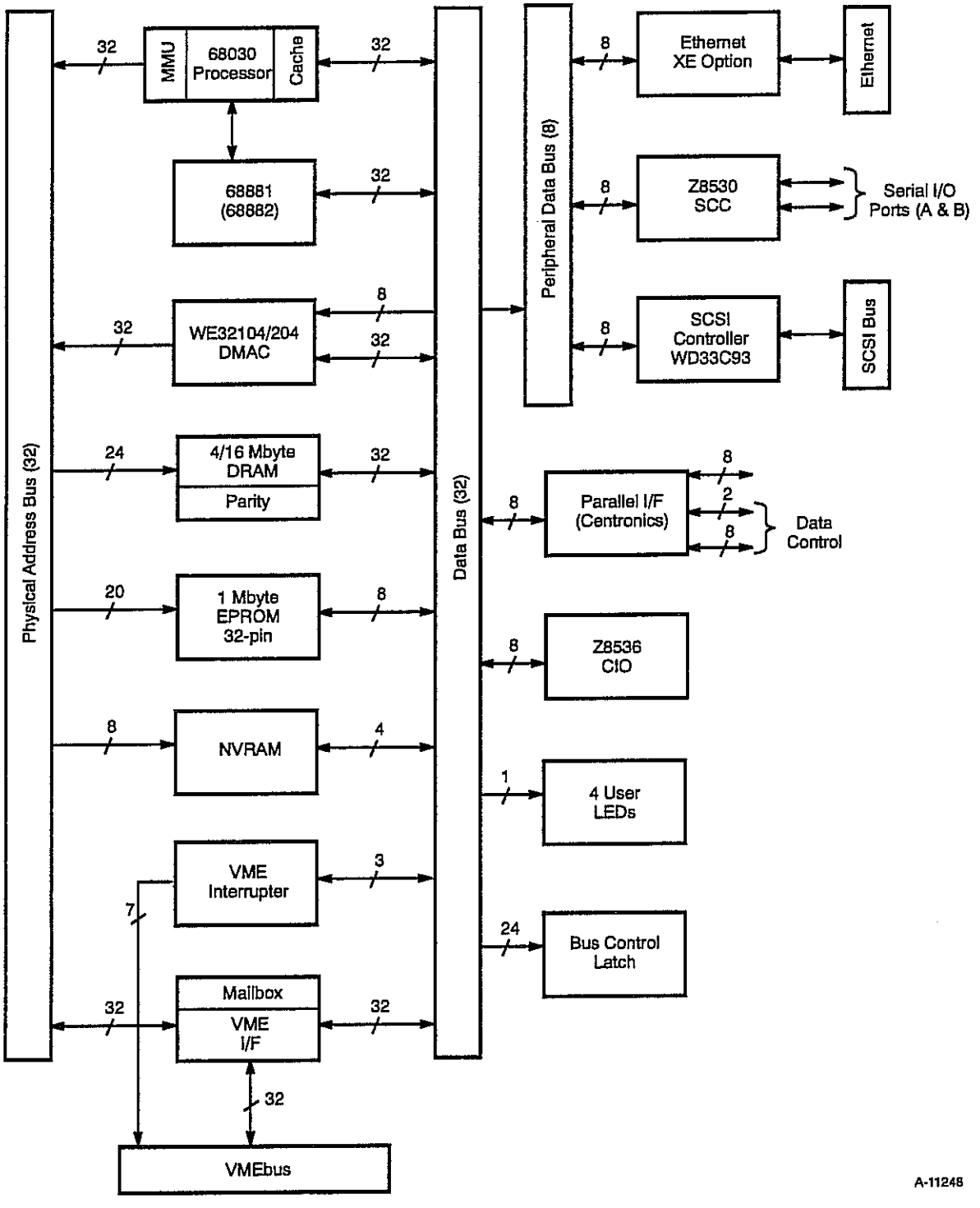


Figure 4-8. HK68/V30 Block Diagram

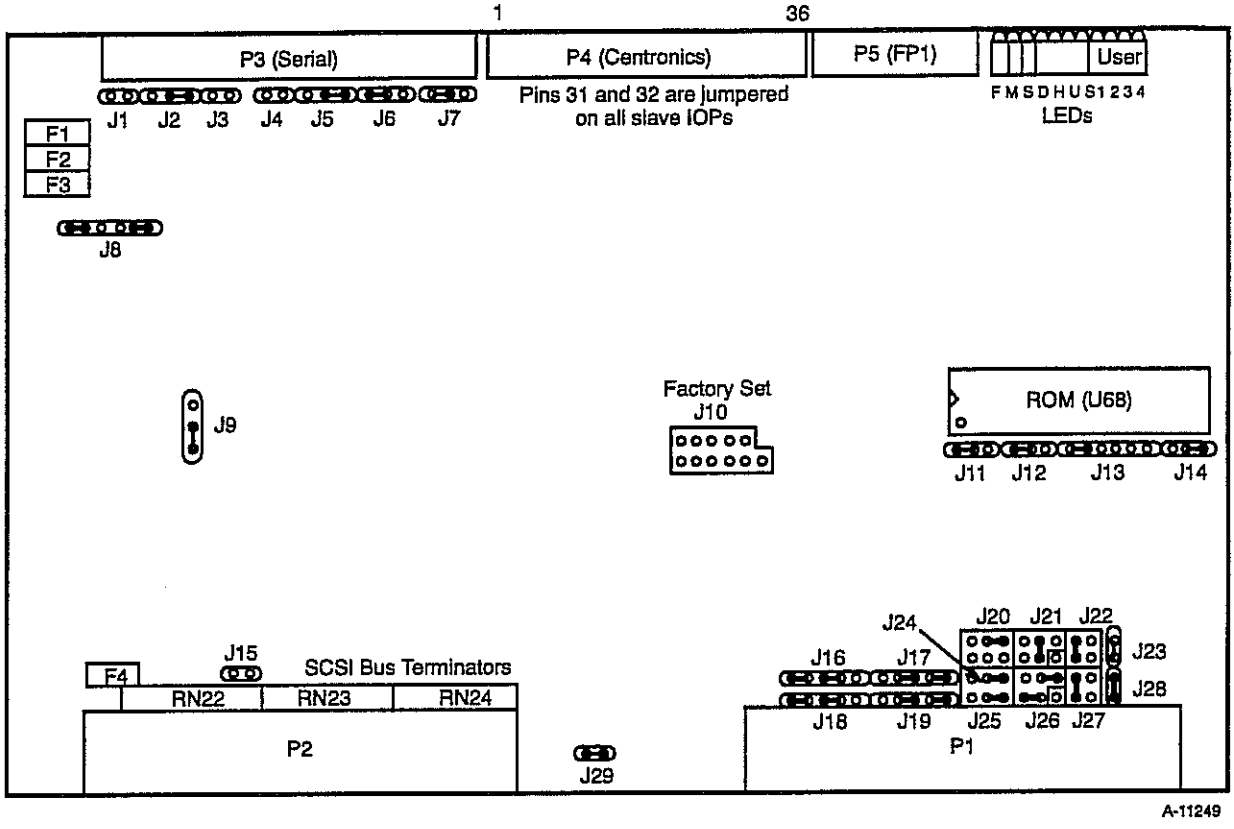


Figure 4-9. Heurikon HK68/V30 Rev. 2 Jumper Locations

DC-4 Disk Drive Controller

The VME I/O controller used to control the DD-4s is manufactured by Xylogics, Inc. The pertinent information about the disk drive controller is as follows:

- Uses a Xylogics model SV7800 IPI-2 to VME disk controller
- Contains multi-threaded 1-Mbyte cache data buffer
- Has two methods of media defect management:
  - One spare sector/track with sector slipping
  - Remap entire track
- Has two status LEDs:
  - Green LED indicates busy
  - Yellow LED – indicates an error – onboard diagnostics failed or a fatal error occurred
- Controls automatic error detection and correction. Uses 64-bit ECC to detect error bursts up to 49 bits long and to correct error bursts up to 17 bits long. It also uses a 16-bit CRC field, appended to each header, to detect error bursts up to 16 bits long.
- Supports a maximum disk data rate of 10 Mbytes/s per IPI channel
- Supports disk drives with data rates between 1.8 Mbytes/s and 10 Mbytes/s
- In CRAY EL series configuration, supports two DD-4 disk drives per IPI channel. There are two IPI channels per DC-4 that enable support of four DD-4 drives.
- Has a DMA data transfer rate of 36 Mbytes/s maximum for longword transfers. Will typically transfer data at a sustained rate of 30 Mbytes/s with no disk transfer in progress and at 20 Mbytes/s when a disk transfer is occurring simultaneously.
- The *Xylogics Disk Controller User's Manual* describes error code and commands in Appendix B.

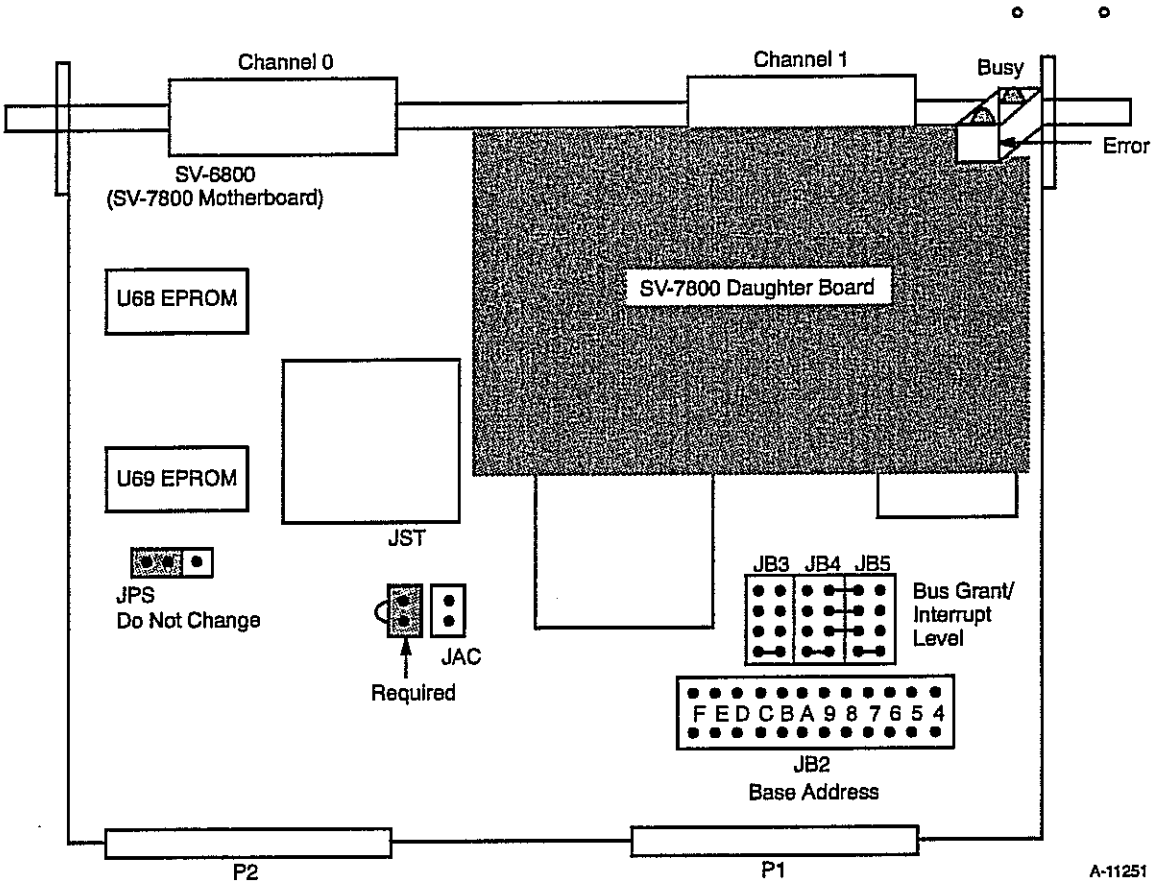


Figure 4-10. SV-7800 Jumper Locations

Controller Address Jumpers for Xylogics SV7800 IPI Controller

The address jumpers are location JB2 on the 7800 controller as shown in Figure 4-11.

Pin key:

- a colon (:) means the jumper is out (maps to a binary 1)
- a vertical bar (|) means the jumper is in (maps to a binary 0)

Address Map:		Jumper Settings:
0xC800 to 0xC8FF	– IPI controller 0	: : | | : | | | | |
0xC900 to 0xC9FF	– IPI controller 1	: : | | : | | : | | |
0xCA00 to 0xCAFF	– IPI controller 2	: : | | : | : | | | |
0xCB00 to 0xCBFF	– IPI controller 3	: : | | : | : : | | |
0xCC00 to 0xCCFF	– IPI controller 4	: : | | : : | | | | |

Address Map:		Jumper Settings:
0xCD00 to 0xCDFF	– IPI controller 5	: : | | : : | : | | | |
0xCE00 to 0xCEFF	– IPI controller 6	: : | | : : : | | | | |
0xCF00 to 0xCFFF	– IPI controller 7	: : | | : : : : | | | |

HYPERchannel Controller HI-1

The pertinent information about the HYPERchannel controller is as follows:

- Uses Network Systems Corporation (NSC) PI 492 processor interface
- Contains 256 Kbytes of memory
- Provides transparent interface from a VMEbus to an NSC data exchange, and then to a HYPERchannel
- The original equipment manufacturer's (OEM) manual contains specifications
- POST LED should illuminate at power-up and remain on for approximately 30 seconds; if illuminated longer, this may indicate a problem on the PI 492.
- Error codes are listed and explained in the OEM manual

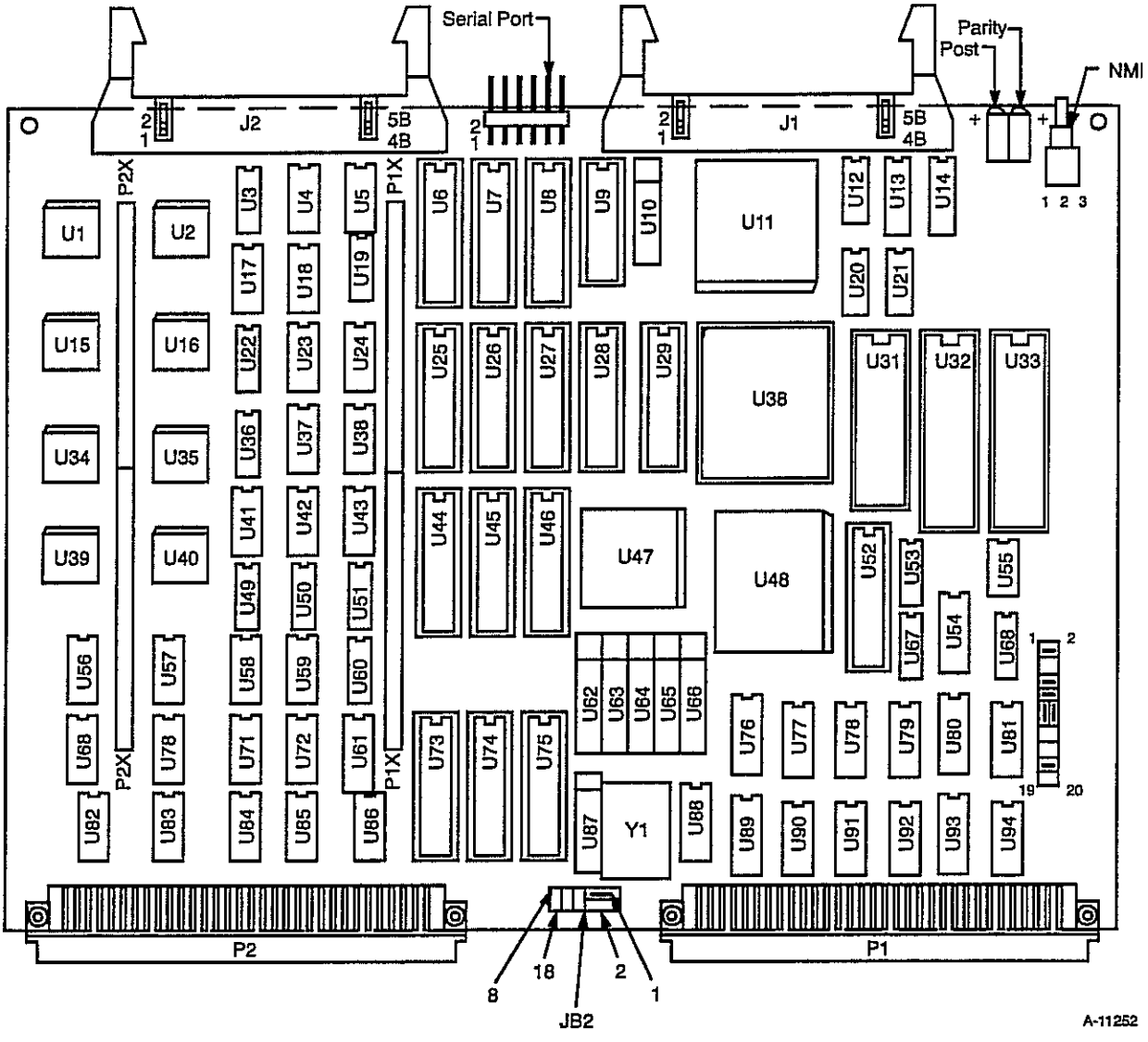


Figure 4-11. PI492 Board Configuration

Table 4-8. Bus Request Level

	Bus Request Level	JB1 Pin Settings
Highest	3 †	1-2, 5-6, 7-8, 9-11, 10-12, 17-18
	2	1-2, 5-6, 7-9, 8-10, 11-12, 15-17
	1	1-2, 3-5, 4-6, 7-8, 11-12, 14-16
	0	1-3, 2-4, 5-6, 7-8, 11-12, 13-14

† CRI uses this level.

Table 4-9. Interrupt Level

	Interrupt Level	JB2 Pin Settings
Highest	7	9-10
	6	7-9
	5	6-8
	4 †	5-6
	3	4-6
	2	1-3
Lowest	1	1-2

† CRI uses this level.

TC-2 Pertec Tape Controller

The pertinent information about the tape drive controller is as follows:

- Is a CIPRICO Tapemaster 3000 (TM3000)
- Contains 4-Kbyte FIFO buffer
- Has maximum bus transfer rates of 10 Mbytes/s
- Has maximum tape transfer rates of 2 Mbytes/s
- Uses DMA to improve data transfer rate
- The OEM product specification manual describes error codes in Appendix C
- Faceplate contains 2 LEDs
  - Green LED indicates busy
  - Red LED indicates error – illuminates when an uncorrectable error is detected
- Requires 200 μ second (maximum) for power-up initialization
- The OEM product specification manual describes specifications

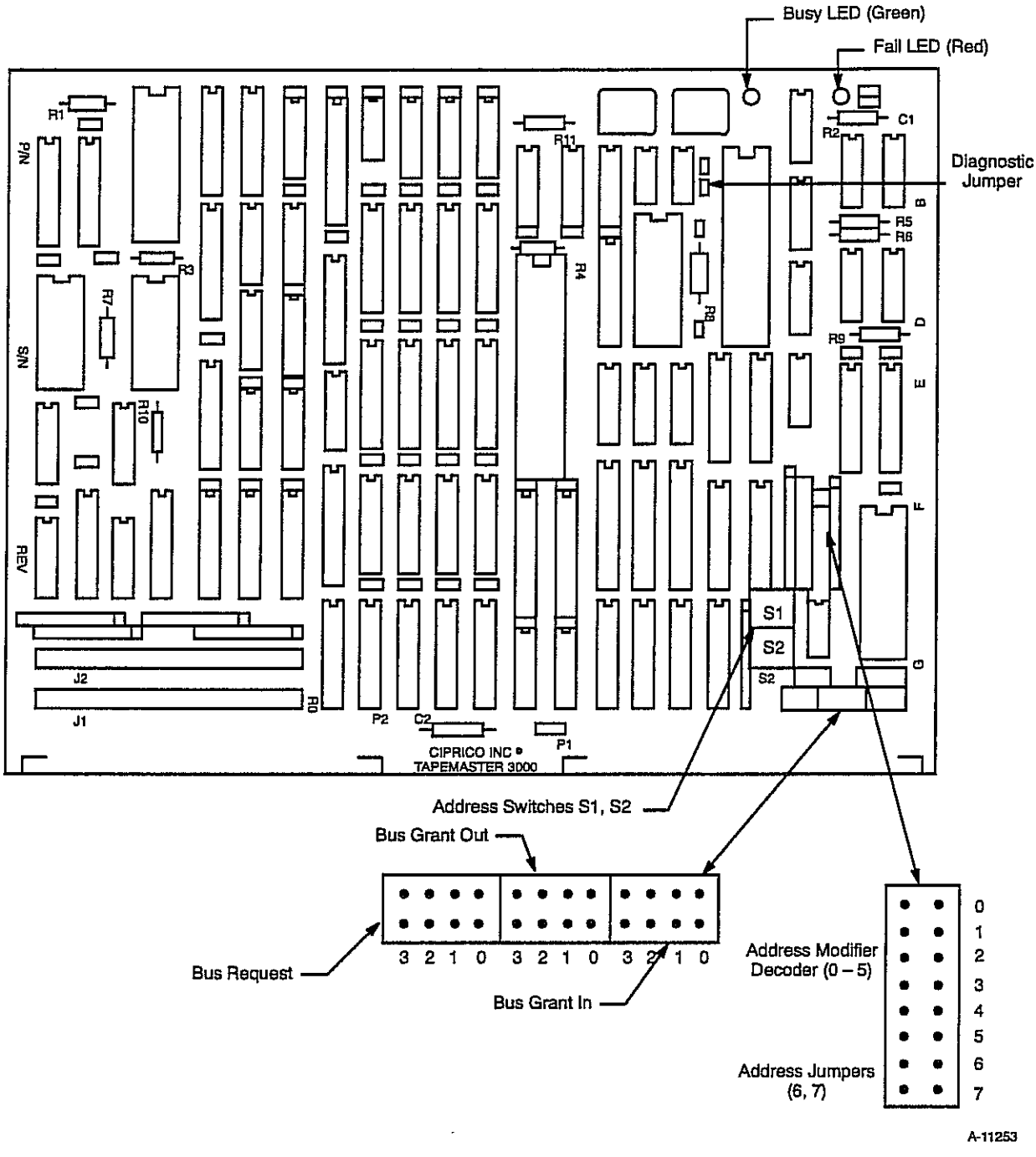


Figure 4-12. Tapemaster 3000 Switch and Jumper Locations

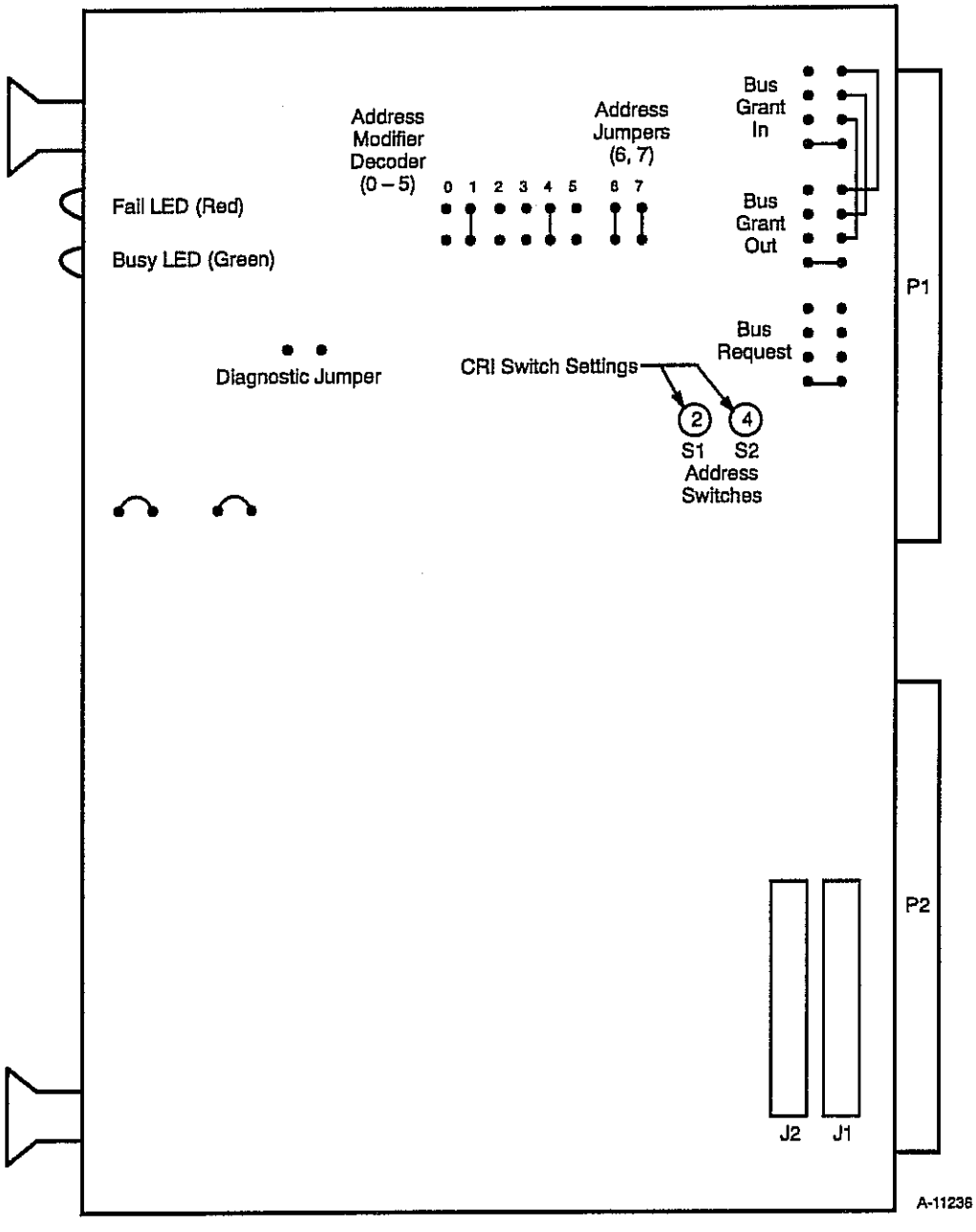


Figure 4-13. TM3000 Jumper Assignments

ESDI Disk Drive Controller DC-3 (CIPRICO Rimfire 3411)

The pertinent information about the disk drive controller is as follows:

- Controls up to 4 ESDI disk drives, full concurrency
- Supports drive data-transfer rates to 20 MHz
- VMEbus transfer rate of 30 Mbytes/s maximum
- Performs data caching on read/write functions
- Sustains cache throughput of 6-Mbytes/s
- Cache is 512 Kbytes of dynamic random-access memory (DRAM) plus a 64-Kbyte static random-access memory (RAM) scratch pad
- Provides both sector and track mapping, sector slip capability; has an onboard defect table
- Uses a CIPRICO proprietary 48-bit error-correction code (ECC)
- Status bit examples are provided in the *CIPRICO Product Specification* manual. These status bits are reported by OLHPA.
- Drive status bit definitions, also returned by OLHPA, are listed in the *CIPRICO Product Specification* manual
- Green LED illuminates when the DC-3 is executing a command
- Red LED illuminates when the DC-3 receives a hard error (cannot be retried or corrected)
- Able to queue up to 50 commands
- On reset, both the red and green LEDs are on and the signal SYSFAIL′ is set. The onboard processor turns off the green LED as soon as possible. The processor then performs initialization and power-on diagnostic checks, which take approximately 5 seconds. When these diagnostics complete, the red LED is turned off and the SYSFAIL′ signal is cleared. The DC-3 is now ready to accept host commands.
- Supports VMEbus block transfers

Refer to the CIPRICO *Product Specification* manual for an explanation of status bits, error status codes, and defect management terms. Refer to Appendix C of the CIPRICO manual for a complete list of error codes.

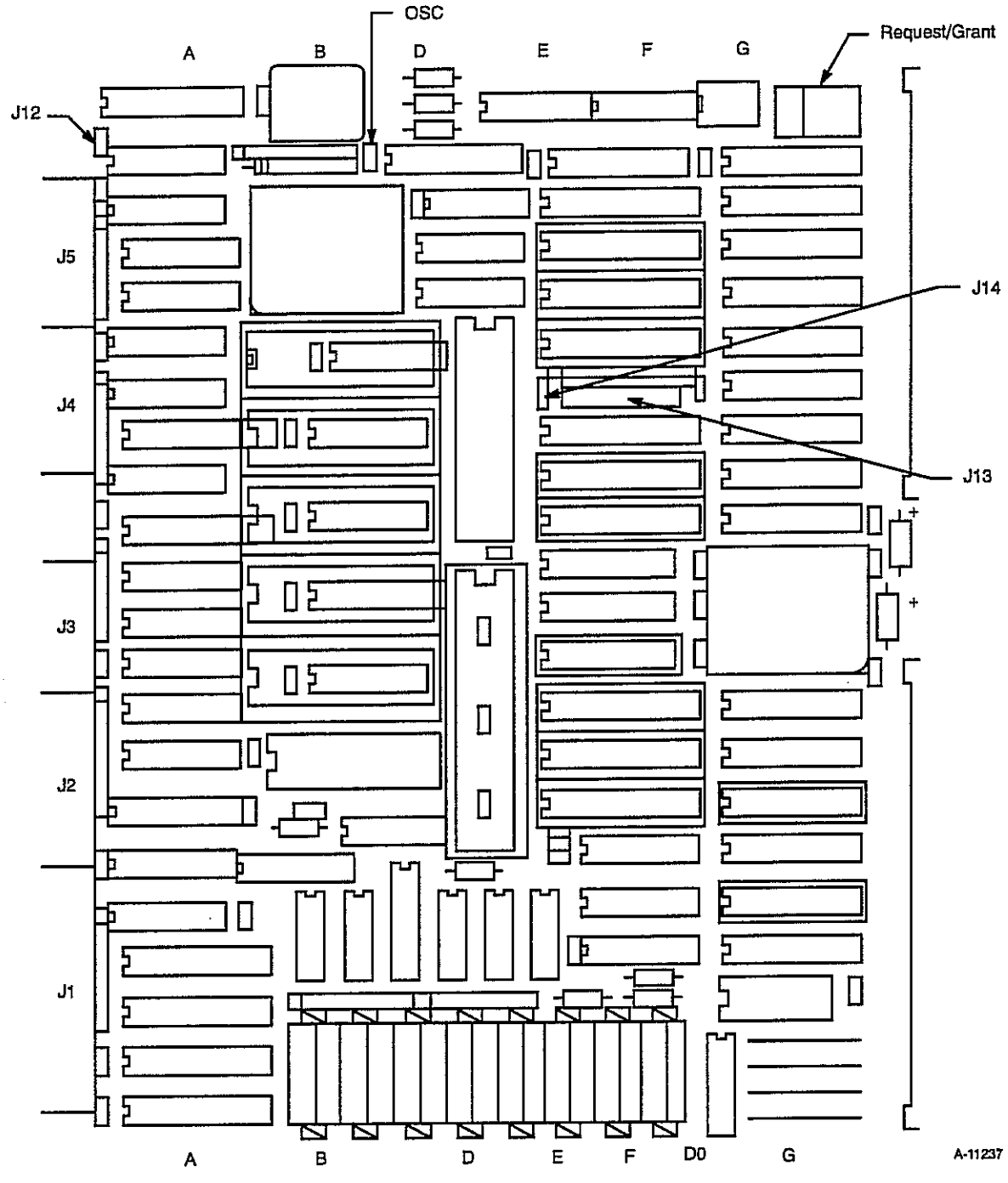


Figure 4-14. Jumper Wire Locations and Designations for CIPRICO RF3411

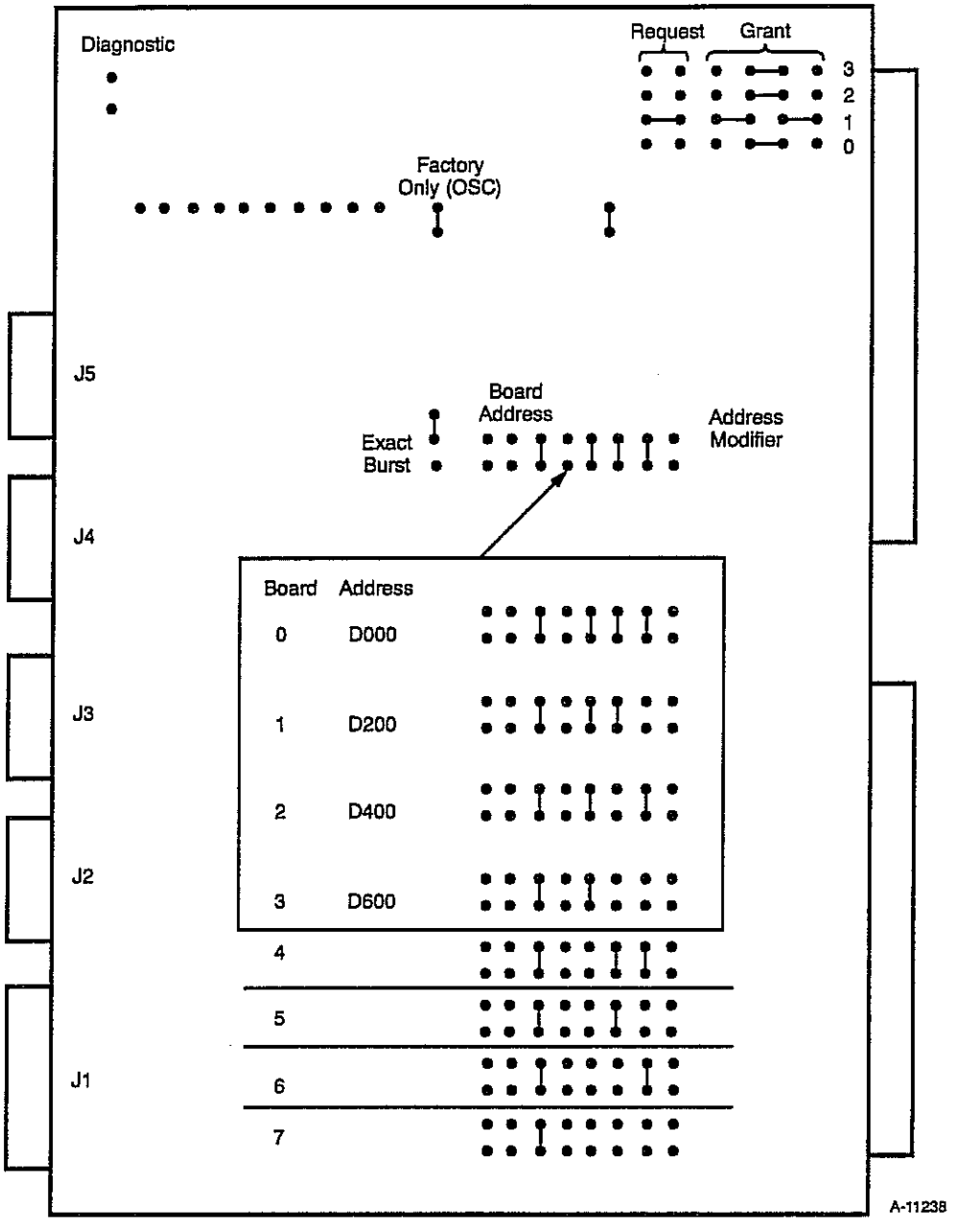


Figure 4-15. ESDI System Disk Controller (CIPRICO RF3411)

CIPRICO Rimfire 3564 (SI-1) SCSI Host Bus Adapter

The pertinent information about the SCSI bus adapter is as follows:

- Adheres to bus protocol required for VME and SCSI
- Acts as an interface between the VMEbus and the SCSI bus
- Uses an NCR 53C94 SCSI controller chip
- Uses a command interface that enables the SI-1 to preprocess a command during a data transfer
- The SCSI port:
  - Supports up to 56 peripheral devices
  - Supports asynchronous 8-bit data up to 2 Mbytes/s, and synchronous 8-bit data up to 5 Mbytes/s
  - Supports full SCSI-2 command set
  - Performs overlapped SCSI operations using disconnect/reconnect
- The VMEbus:
  - Supports data transfer rates of 20 Mbytes/s burst and 30 Mbytes/s in block mode
  - Supports 16-, 24-, and 32-bit VMEbus addressing
- Uses an Intel Corporation 16-MHz 80186 microprocessor
- Has direct memory access (DMA) capability through the CIPRICO short burst FIFO (SBF)
- The *CIPRICO RF3560 User's Guide* lists power-on self-test error codes
- 2 LEDs
  - Green LED indicates that the controller is busy
  - Red LED indicates an error or power-on in progress

Refer to *CIPRICO RF3560 User's Guide* for correct addressing.

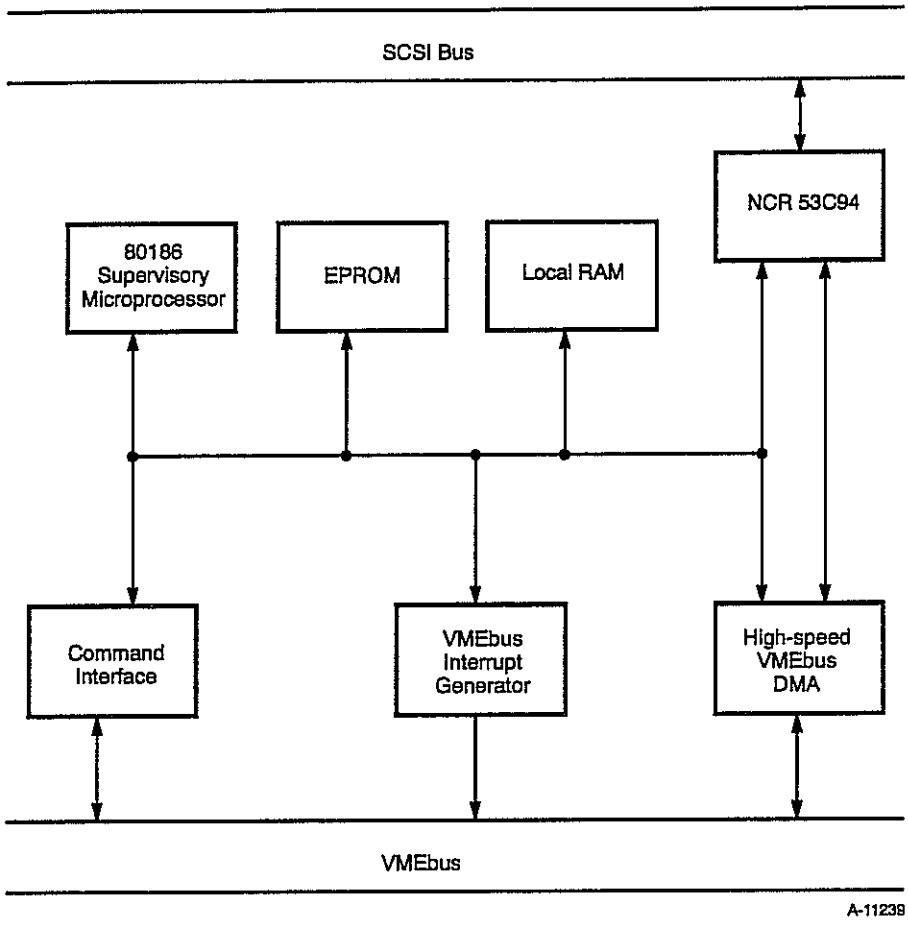


Figure 4-16. Block Diagram of Rimfire RF3564

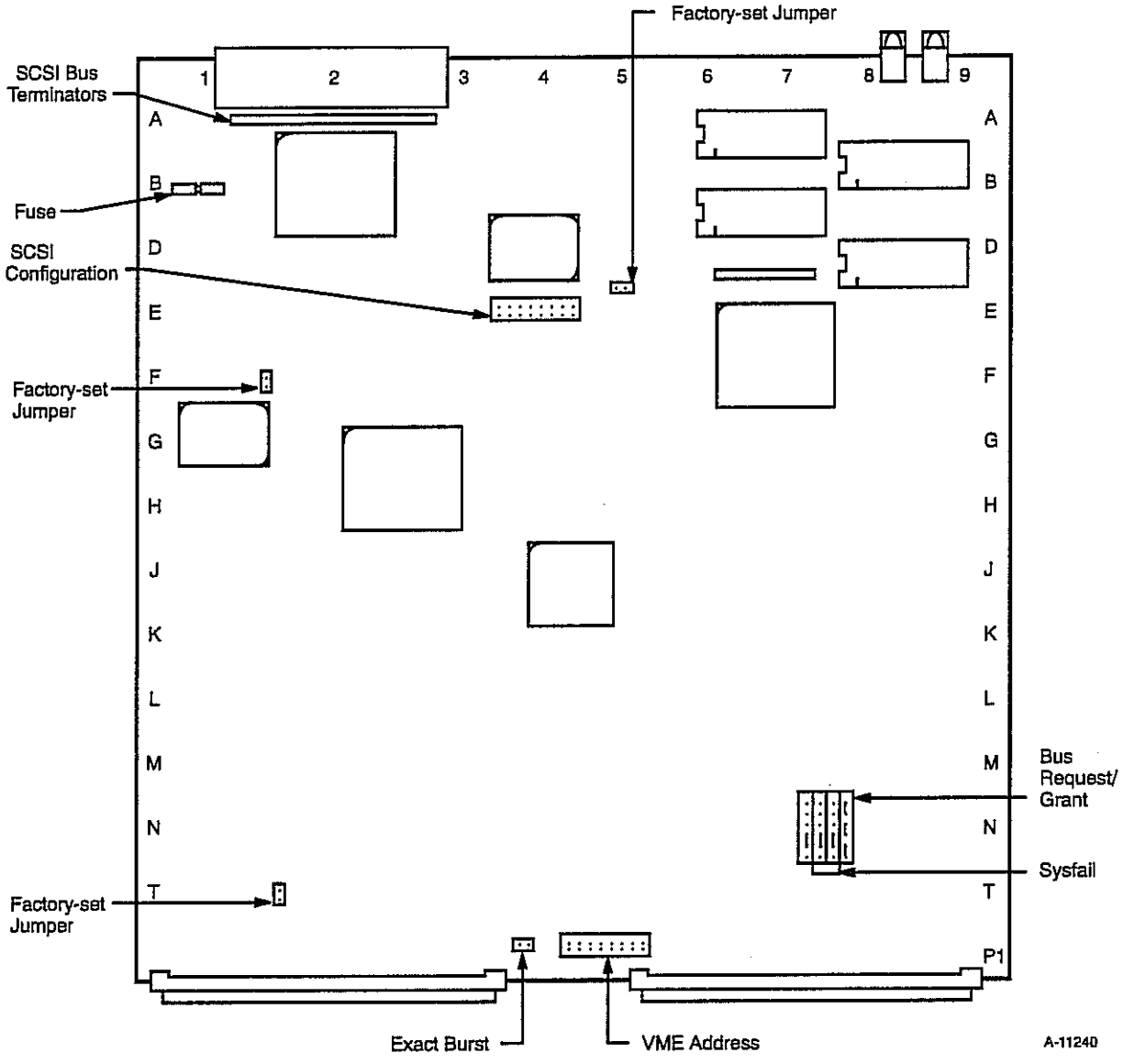
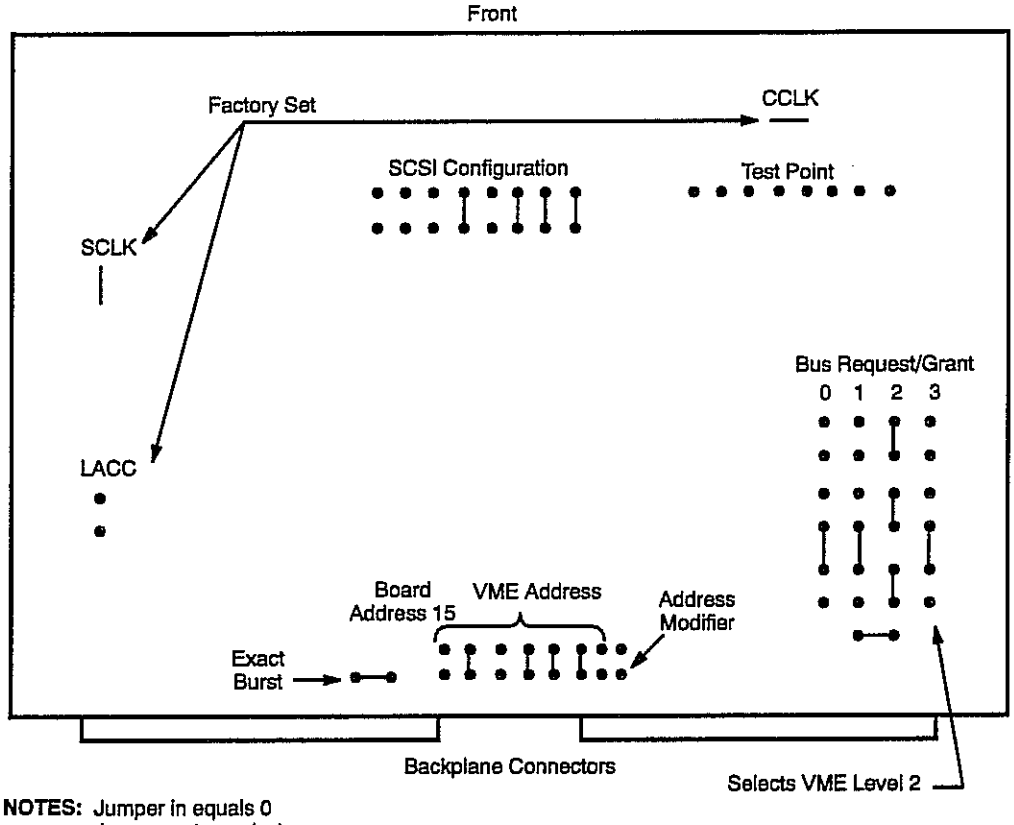


Figure 4-17. Locations of Rimfire RF3564 Jumpers



NOTES: Jumper in equals 0
Jumper out equals 1
Address is set to A000h
Address modifier is set to 2Dh

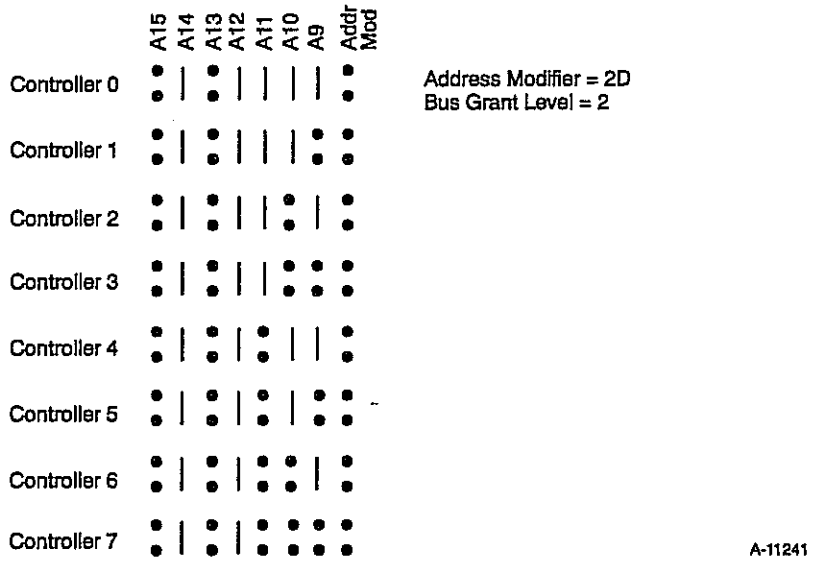


Figure 4-18. CIPRICO 3564 SCSI Controller

Interphase Corporation 4220 Cougar (SI-2) SCSI Host Bus Adapter

The SI-2 controller provides the interface between the IOS and the DD-5S system disk drives. The SI-2 has two fast and narrow SCSI-2 channels; each channel supports synchronous SCSI transfers at a rate of 10 Mbytes/s. Additional SI-2 controllers can be installed to control external SCSI devices. Refer to Figure 4-19 for board jumper configurations.

NOTE: Refer to the Interphase Corporation OEM manual, CRI publication number CZM-0869-000, for SI-2 board jumper setting descriptions and jumper block pin designations. (Contact Logistics or your Cray Research account manager for details on ordering OEM manuals.)

Beta versions of CRAY EL series systems include a DC-3 controller for DD-3 disk drives.

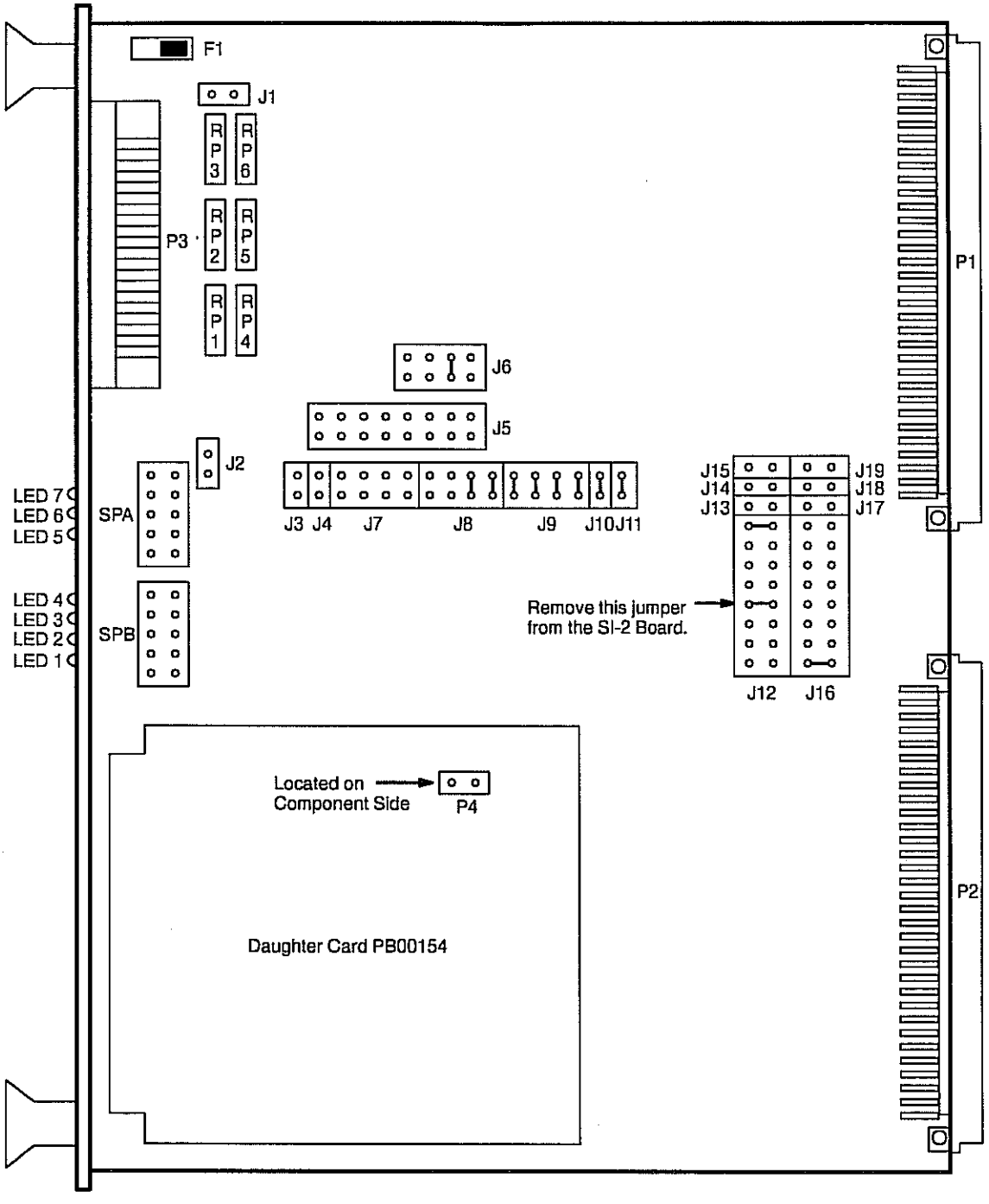


Figure 4-19. SI-2 Board Jumper Configurations

Interphase Corporation V/FDDI Peregrine FI-1 and FI-2

Customers may opt to equip the VME IOS with a fiber distributed data interface (FDDI) designated FI-1 or FI-2. The FDDI supports either single or dual media access control (MAC) in compliance with the FDDI standard. Refer to Figure 4-20 and Figure 4-21, respectively, for the FI-1 and FI-2 jumper configurations.

NOTE: Refer to the Interphase Corporation OEM manuals, Cray Research publication numbers CZM-0888-000 and CZM-0864-000, respectively, for FI-1 and FI-2 board jumper setting descriptions and jumper block pin designations. (Contact Logistics or your Cray Research account manager for details on ordering OEM manuals.)

Each FDDI provides a 100-Mbit/s FDDI network to the VMEbus interface and includes a 1-Mbyte buffer to eliminate data overruns. Both FI-1 and FI-2 have a 512-Kbyte execution random-access memory (RAM) and a VMEbus transfer rate of 30 Mbytes/s.

A common boot interface within the FDDI accepts host-generated diagnostics and generates common boot error messages. No error messages are generated other than FAIL (ASCII values); the red LED (LED 1) remains illuminated.

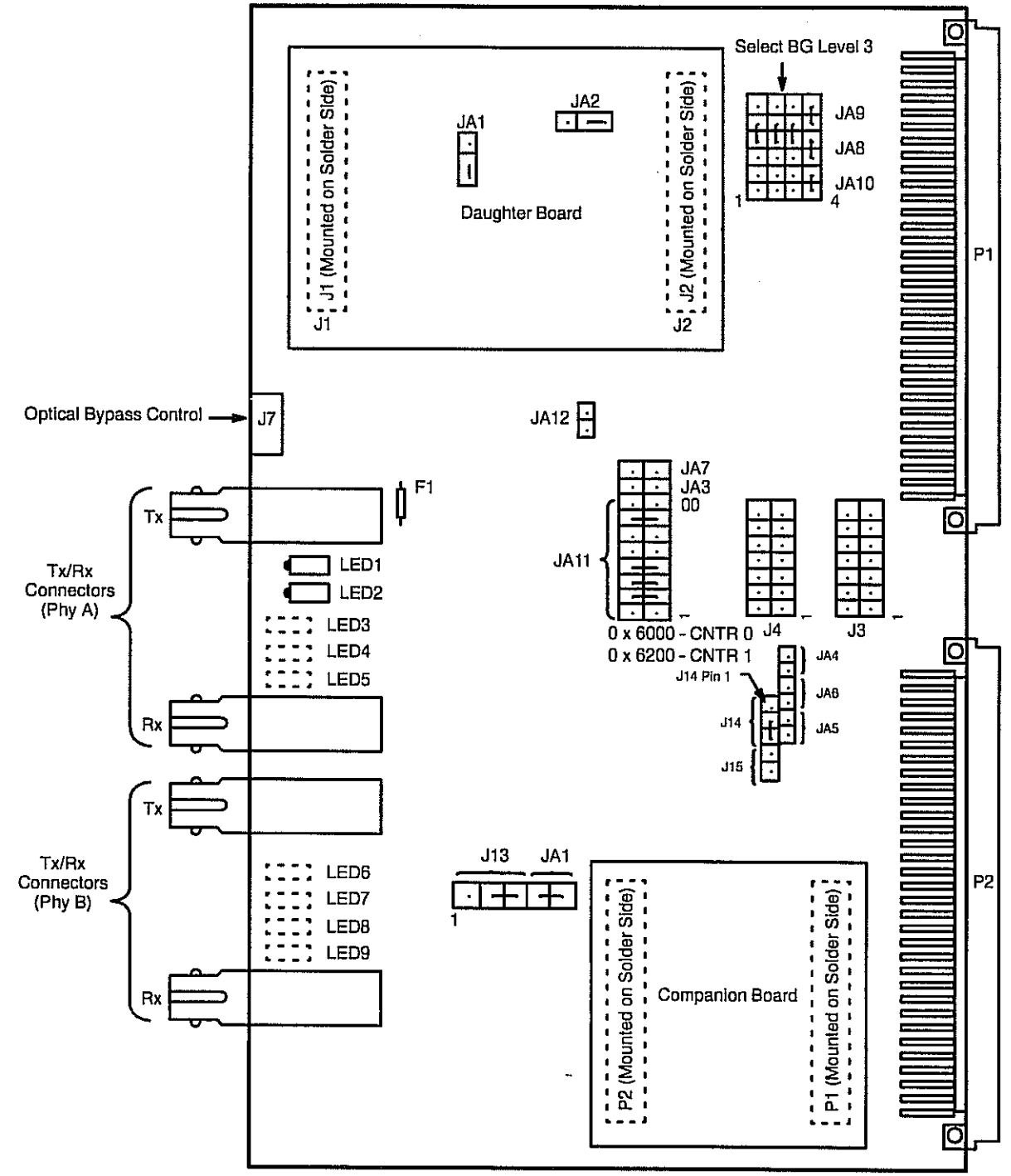


Figure 4-20. FI-1 Board Jumper Configurations

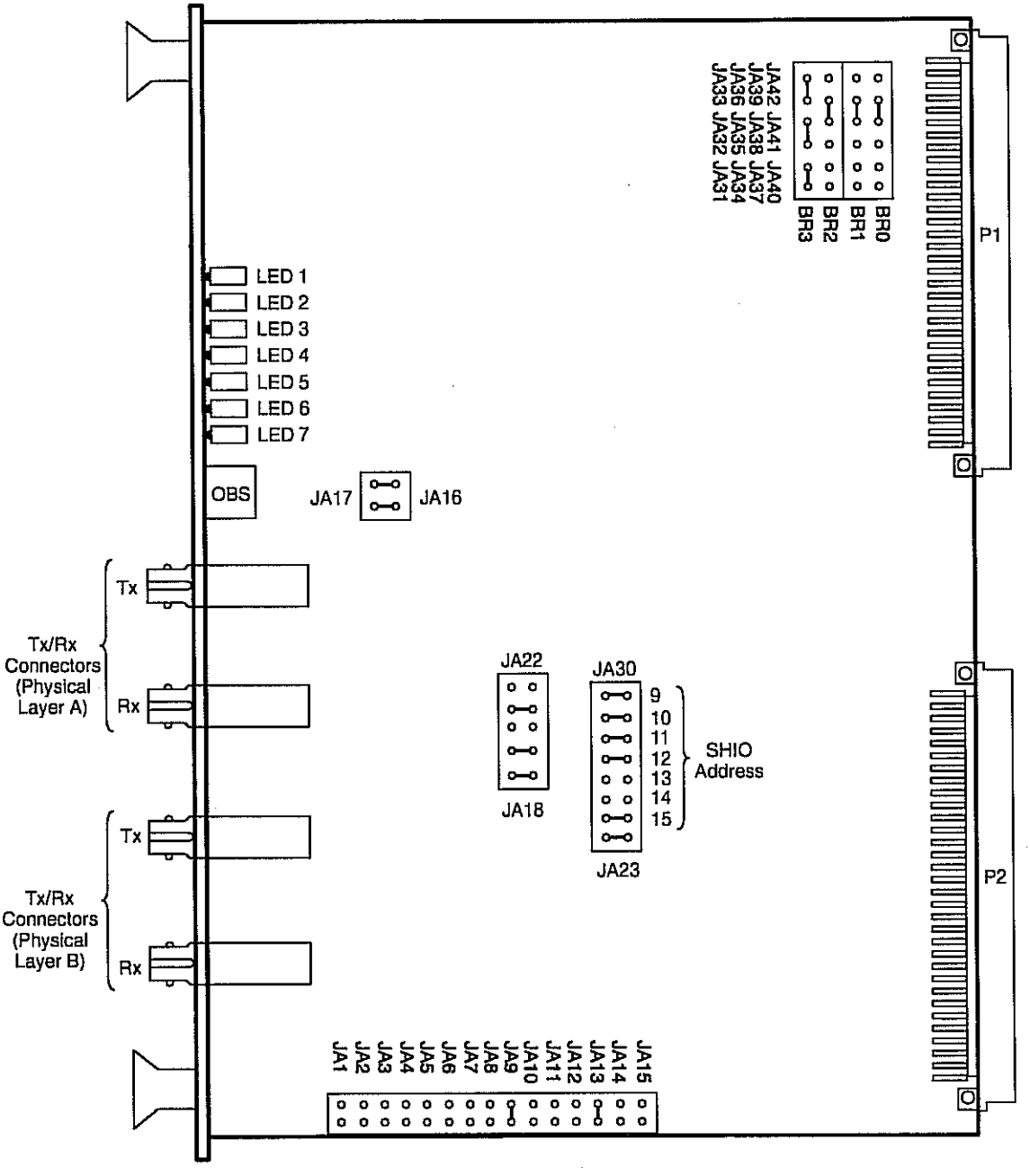


Figure 4-21. FI-2 Board Jumper Configurations

CMC-130 Series Ethernet Controller EI-1

The EI-1 controller provides an interface between the VME IOS and a customer Ethernet network. The EI-1 controller conforms to the 10-Mbyte/s carrier sense multiple access/collision detection (CSMA/CD) standard. CRAY EL series systems may contain either of two board designs; refer to Figure 4-22 (previous design) and Figure 4-23 (revised design) for the EI-1 board jumper configurations.

NOTE: Refer to the CMC Corporation OEM manual, CRI publication number CZM-0865-000, for EI-1 board jumper setting descriptions and jumper block pin designations. (Contact Logistics or your Cray Research account manager for details on ordering OEM manuals.)

The EI-1 controller contains a complete microprocessor (MC68020) and has 256-Kbyte onboard dynamic random-access memory (DRAM) with 256-Kbyte global data video dynamic random-access memory (VDRAM). The EI-1 controller runs self-tests upon power-up; if a failure occurs, the red LED remains illuminated.

Refer to the *CMC Reference Guide* for error descriptions.

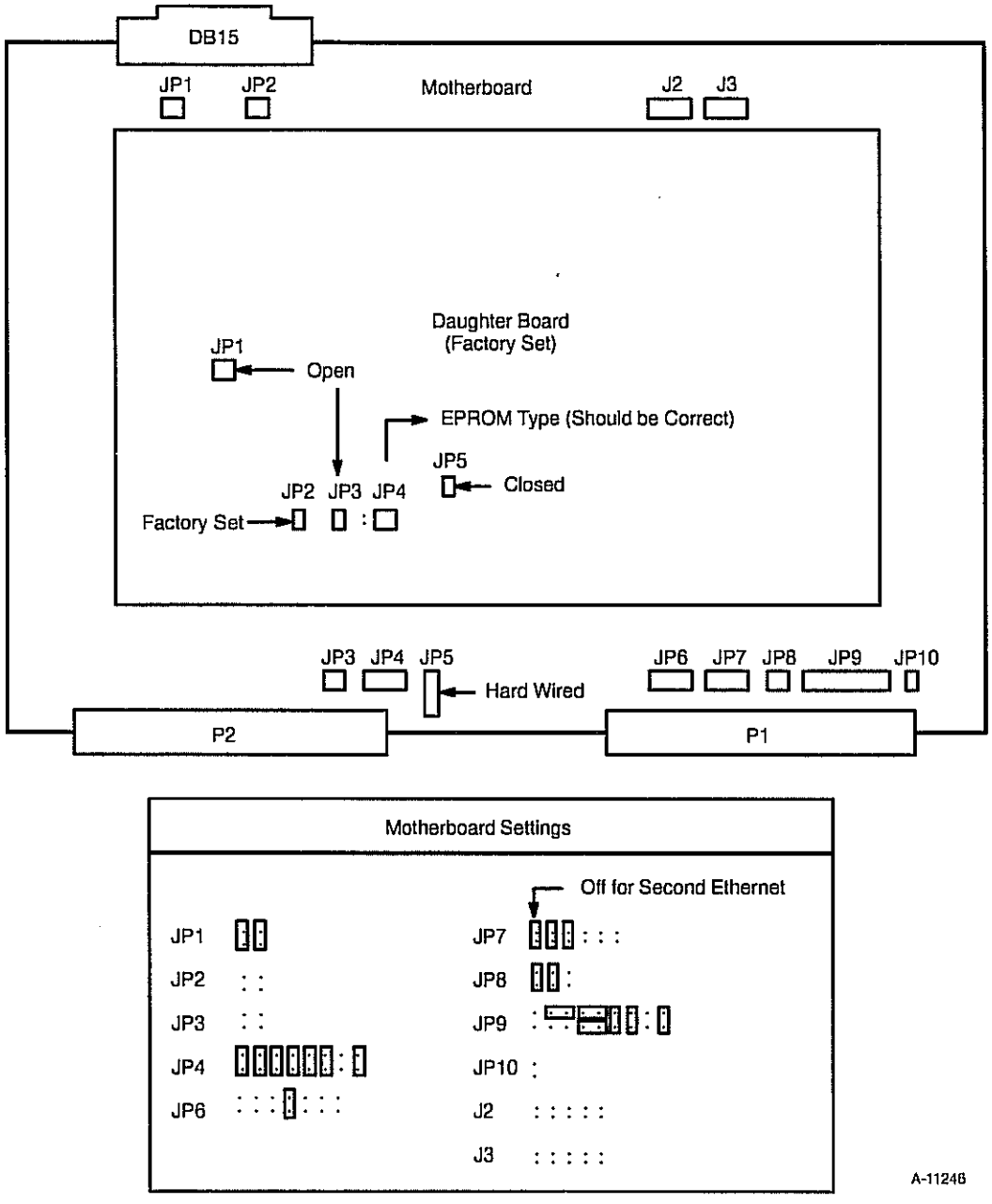


Figure 4-22. Previous Design EI-1 Board Jumper Configurations

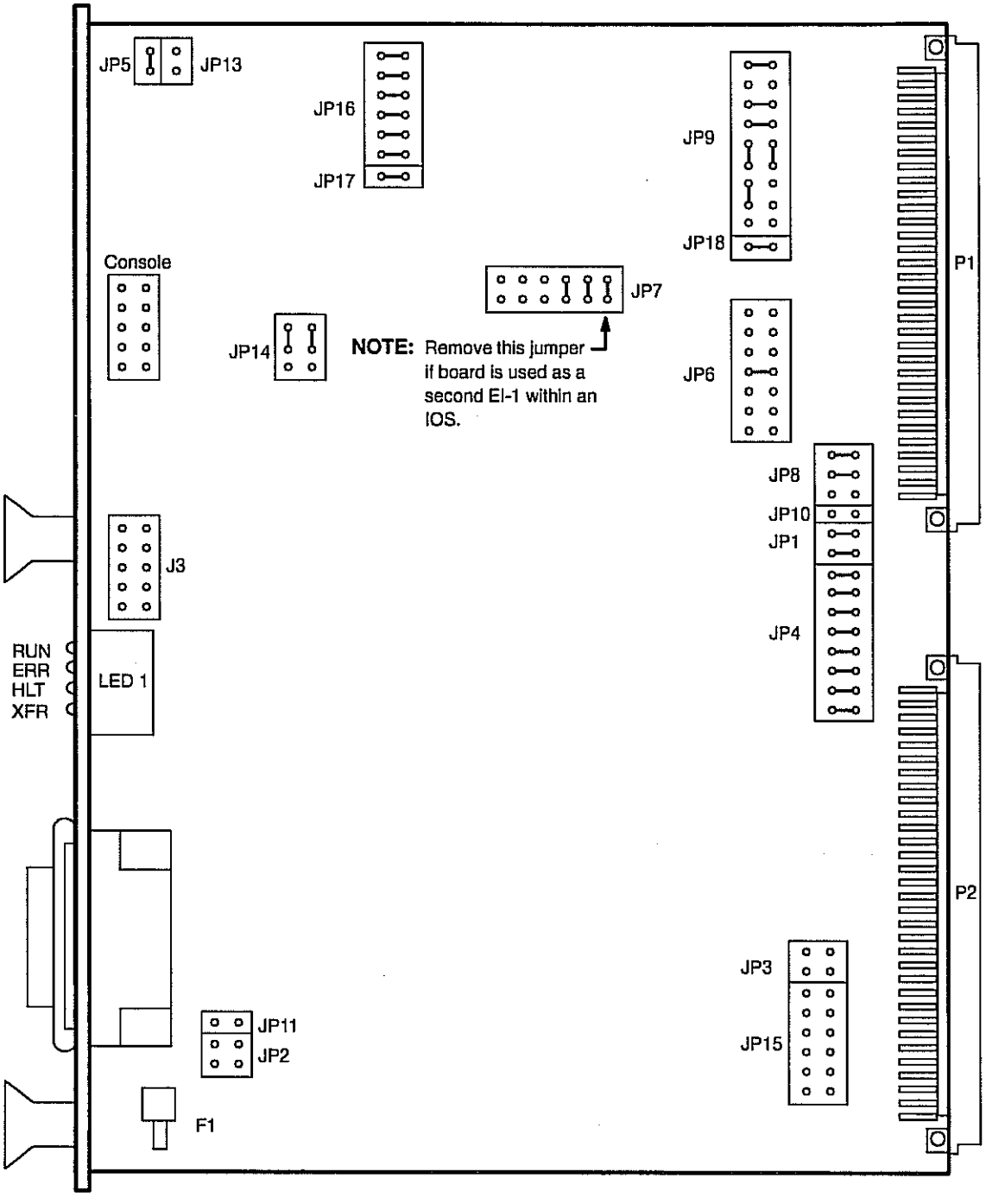


Figure 4-23. Revised Design EI-1 Board Jumper Configurations

Maximum Strategy, Inc. Strategy 2 Disk Array Controller (DAC)

The pertinent information about the DAC is as follows:

- Handles all details of disk flaw mapping, error detection, and error correction
- Supports up to four high-speed data channels

Refer to the *Maximum Strategy Inc. Disk Array Controller Operation* manual, Revision C5, for format command information.

NOTE: The DAC will not be available on the CRAY EL98.

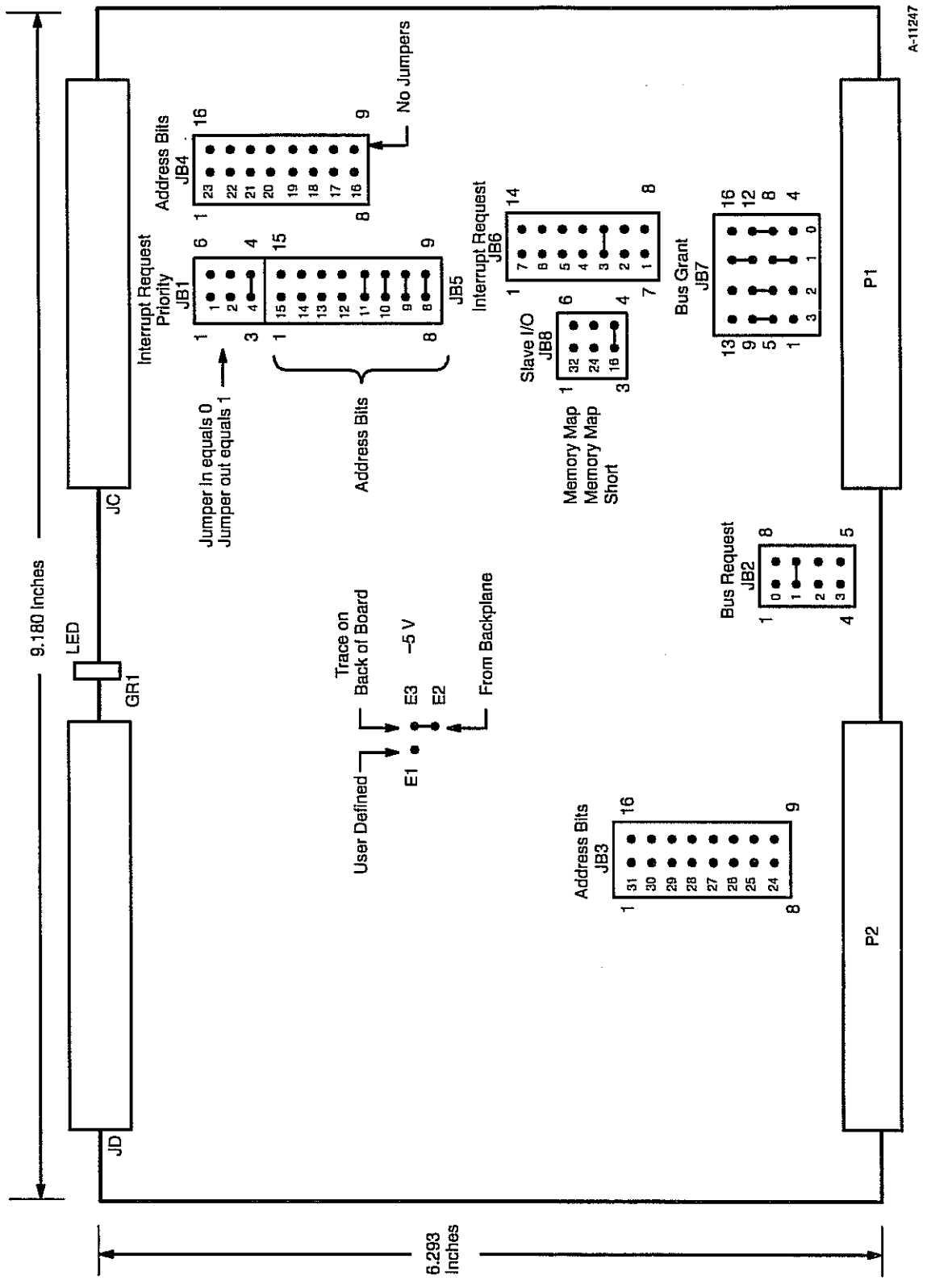


Figure 4-24. VME Array Controller Configuration (Bus Master)

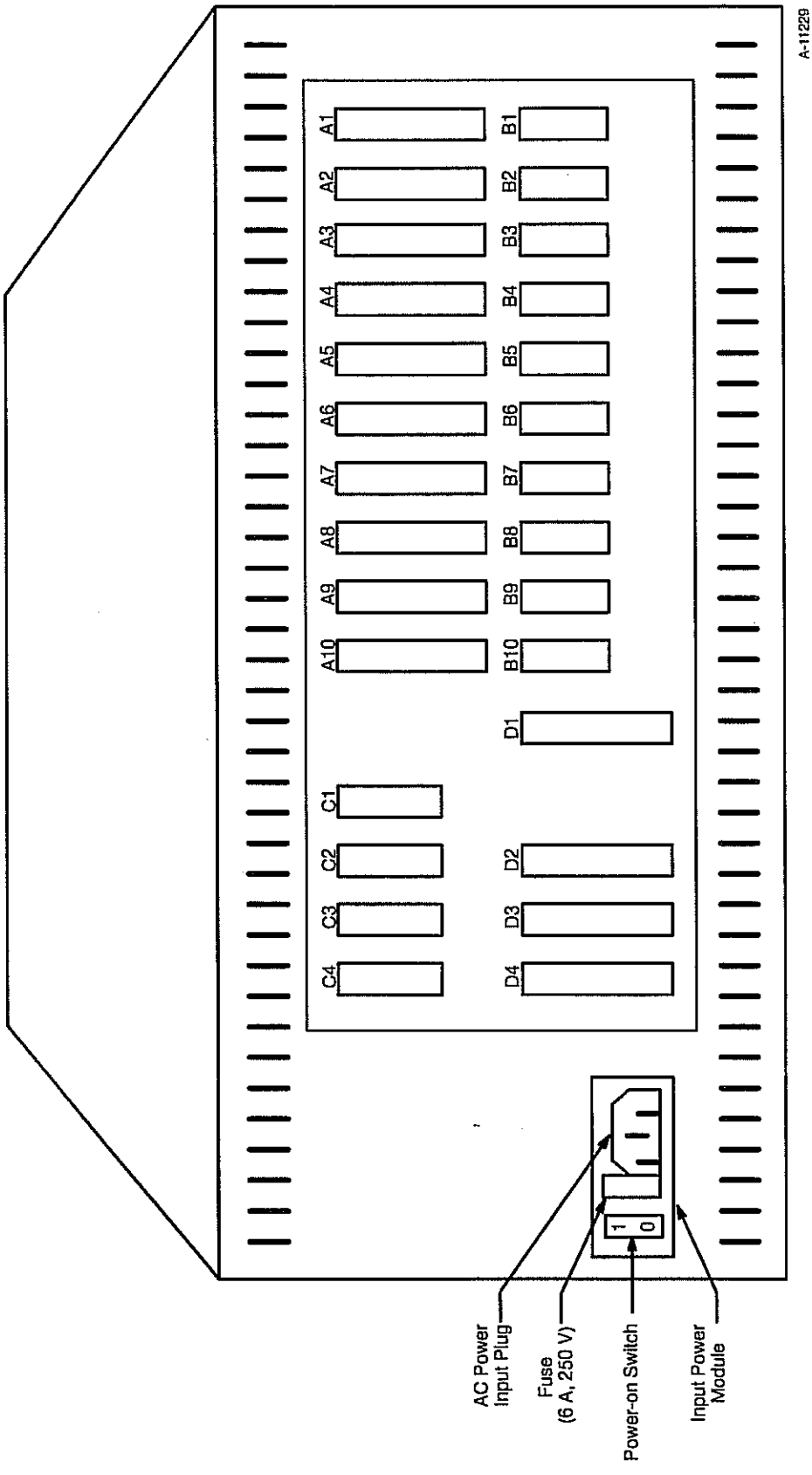
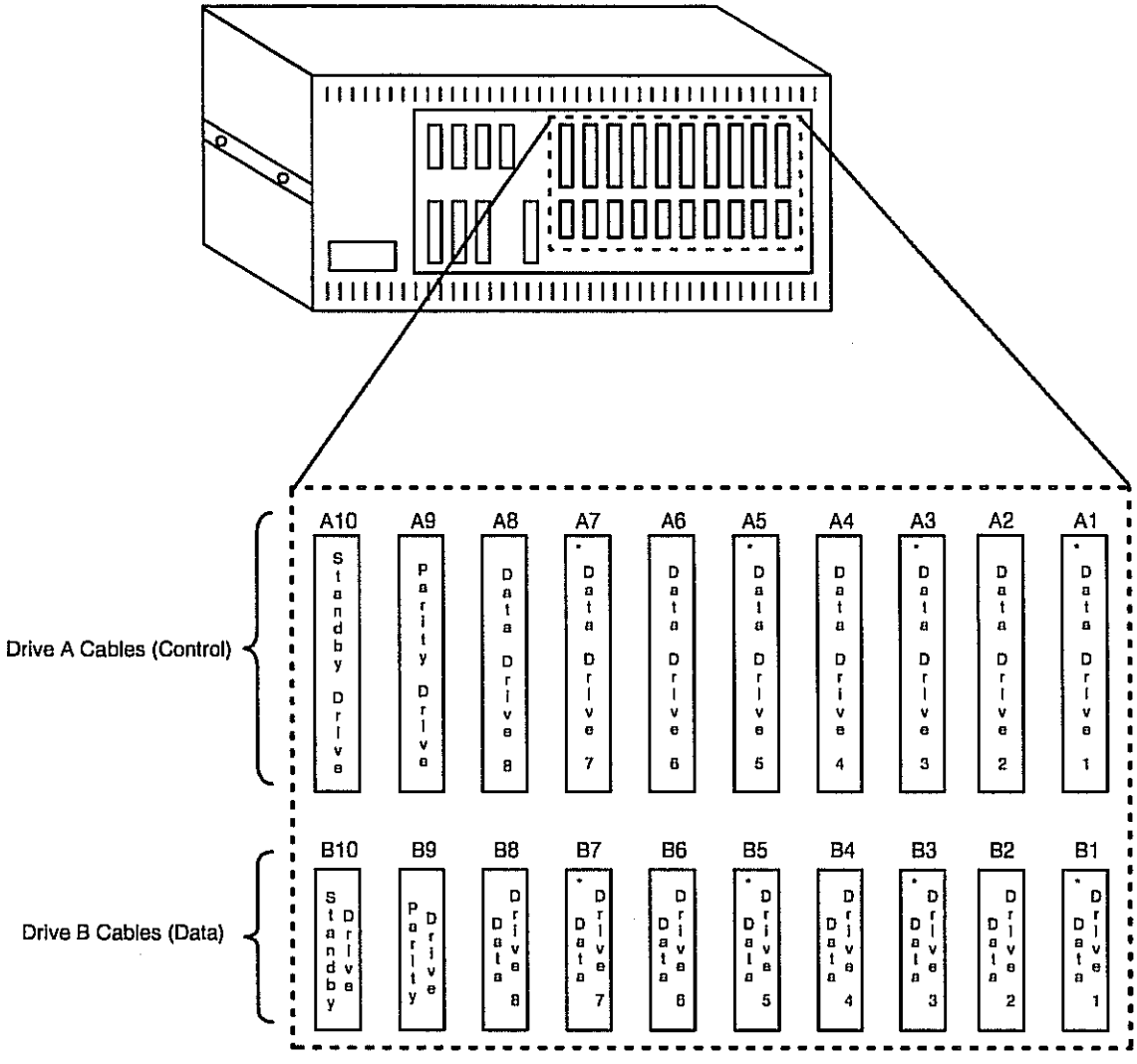


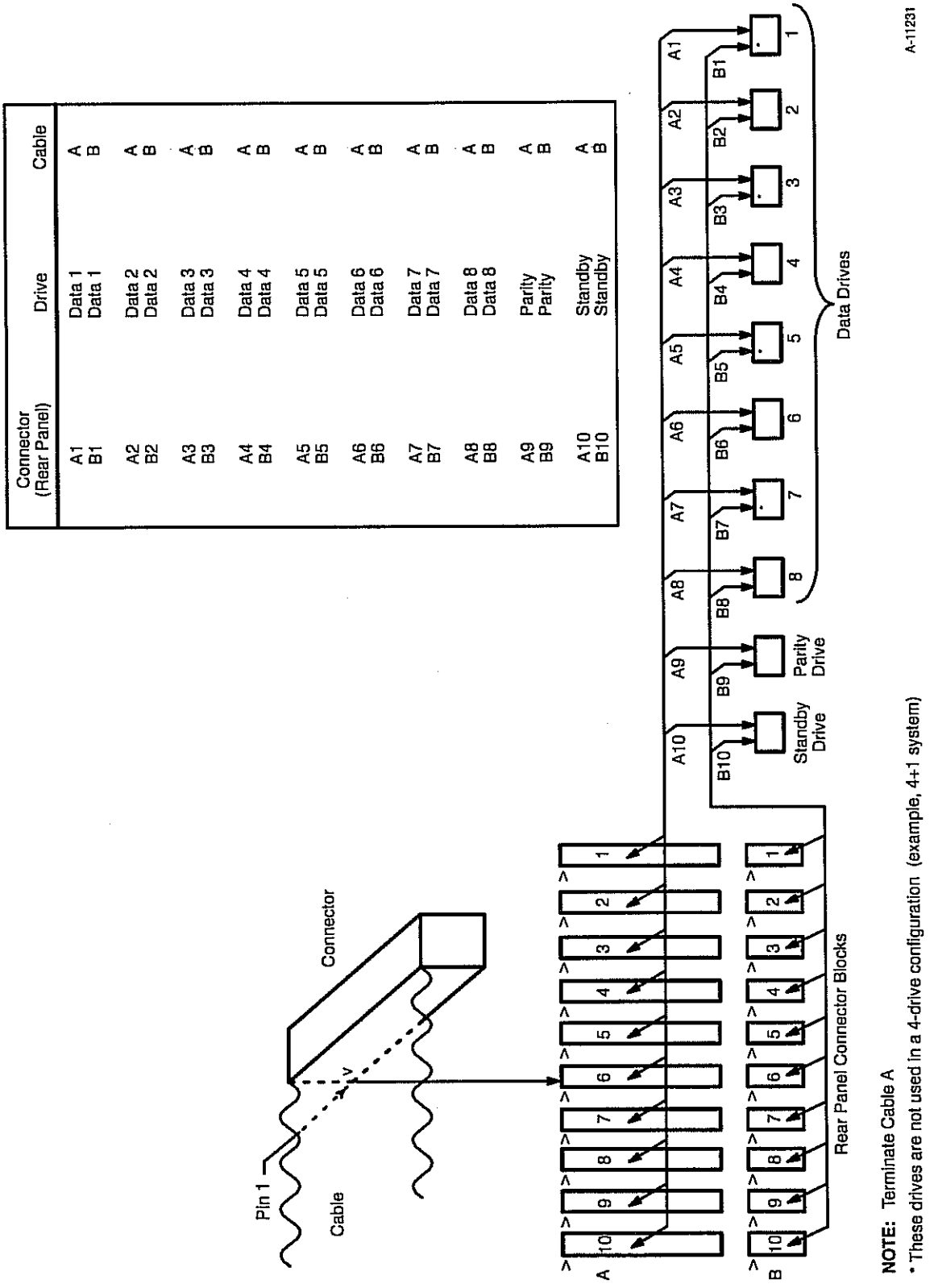
Figure 4-25. 14-inch Rear Chassis Panel, DAC Chassis



NOTE: Connector blocks marked with asterisks (*) are not used in a configuration of four data drives.

A-11230

Figure 4-26. Drive Connector Block Cable Configuration (14-inch Chassis)



NOTE: Terminate Cable A

* These drives are not used in a 4-drive configuration (example, 4+1 system)

Figure 4-27. Drive Interface Cabling (14-inch Chassis)

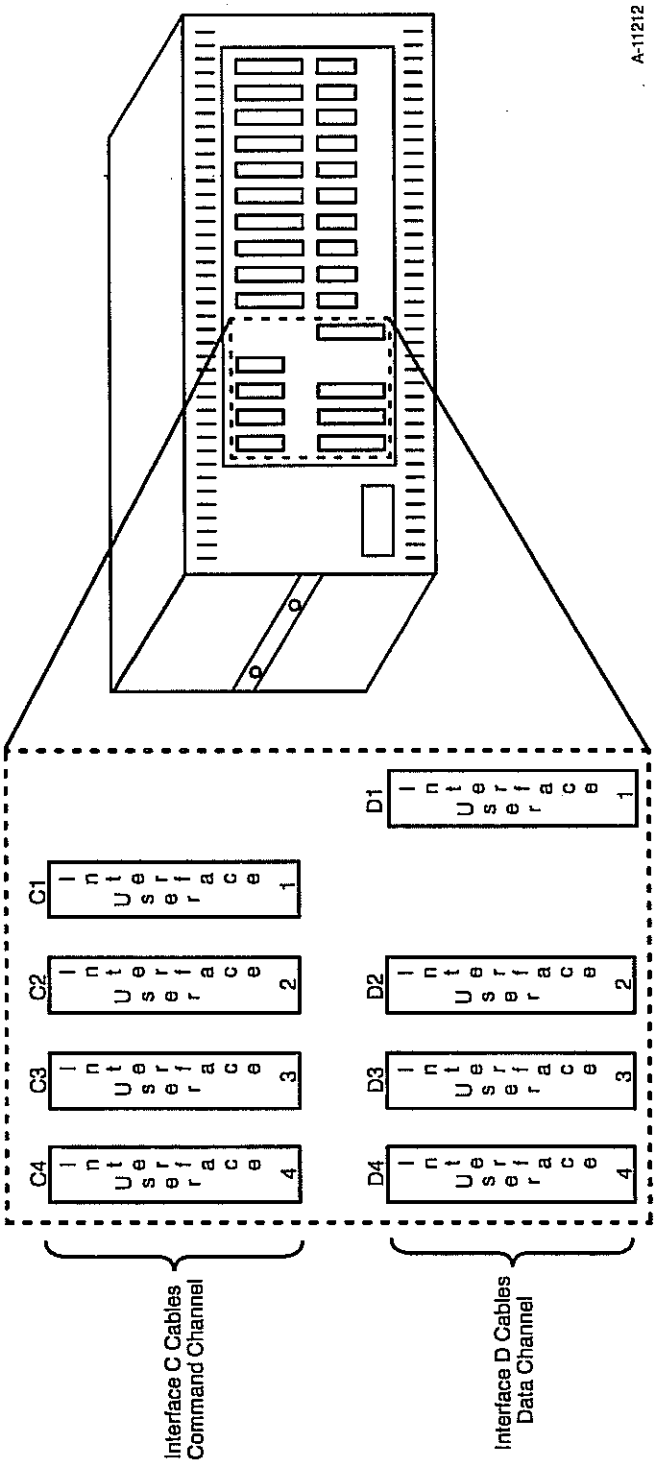


Figure 4-28. User Interface Connector Blocks (14-inch Chassis)

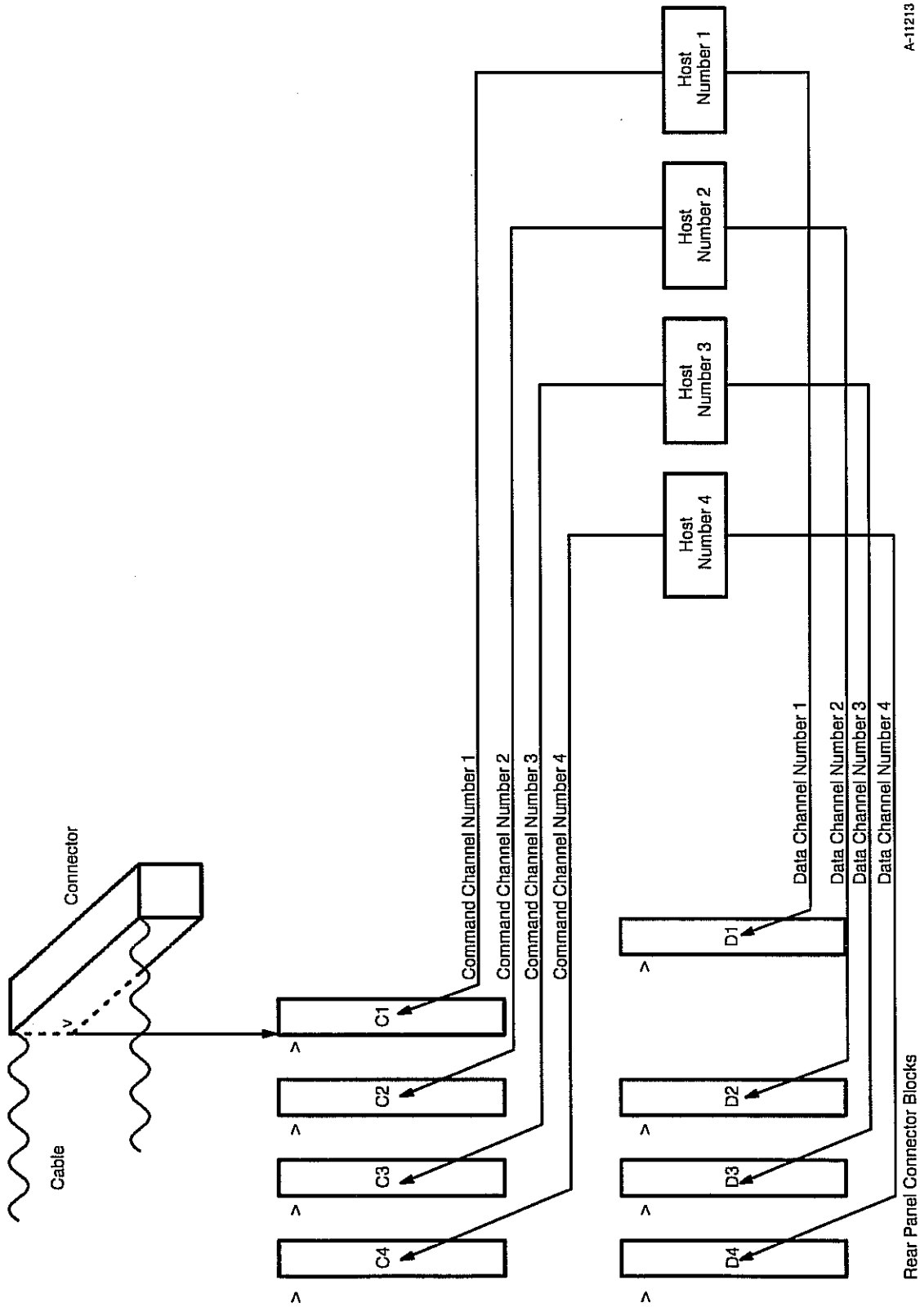


Figure 4-29. User Interface Cabling (14-inch Chassis)

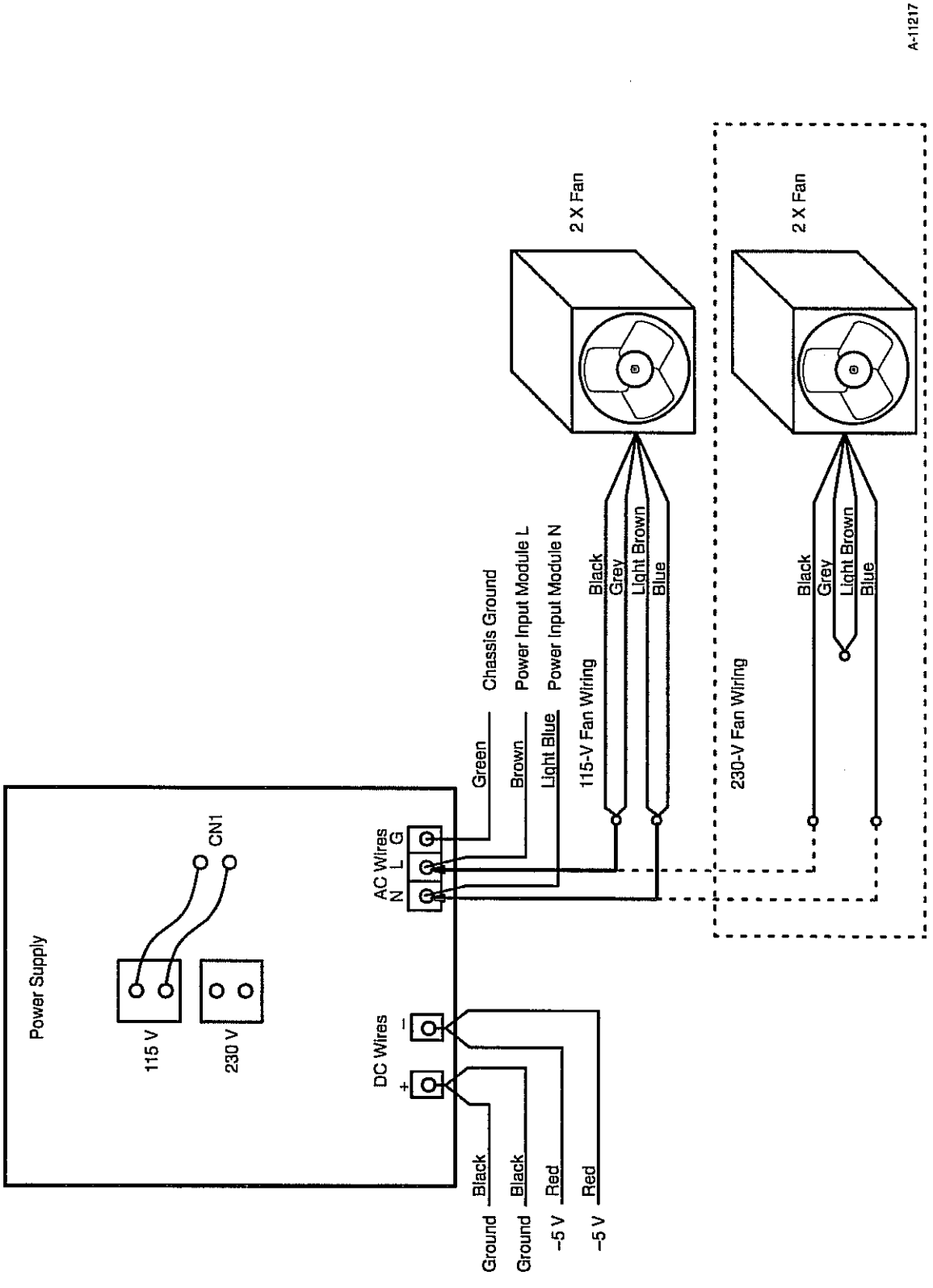


Figure 4-30. AB Power Supply and Fan Wiring

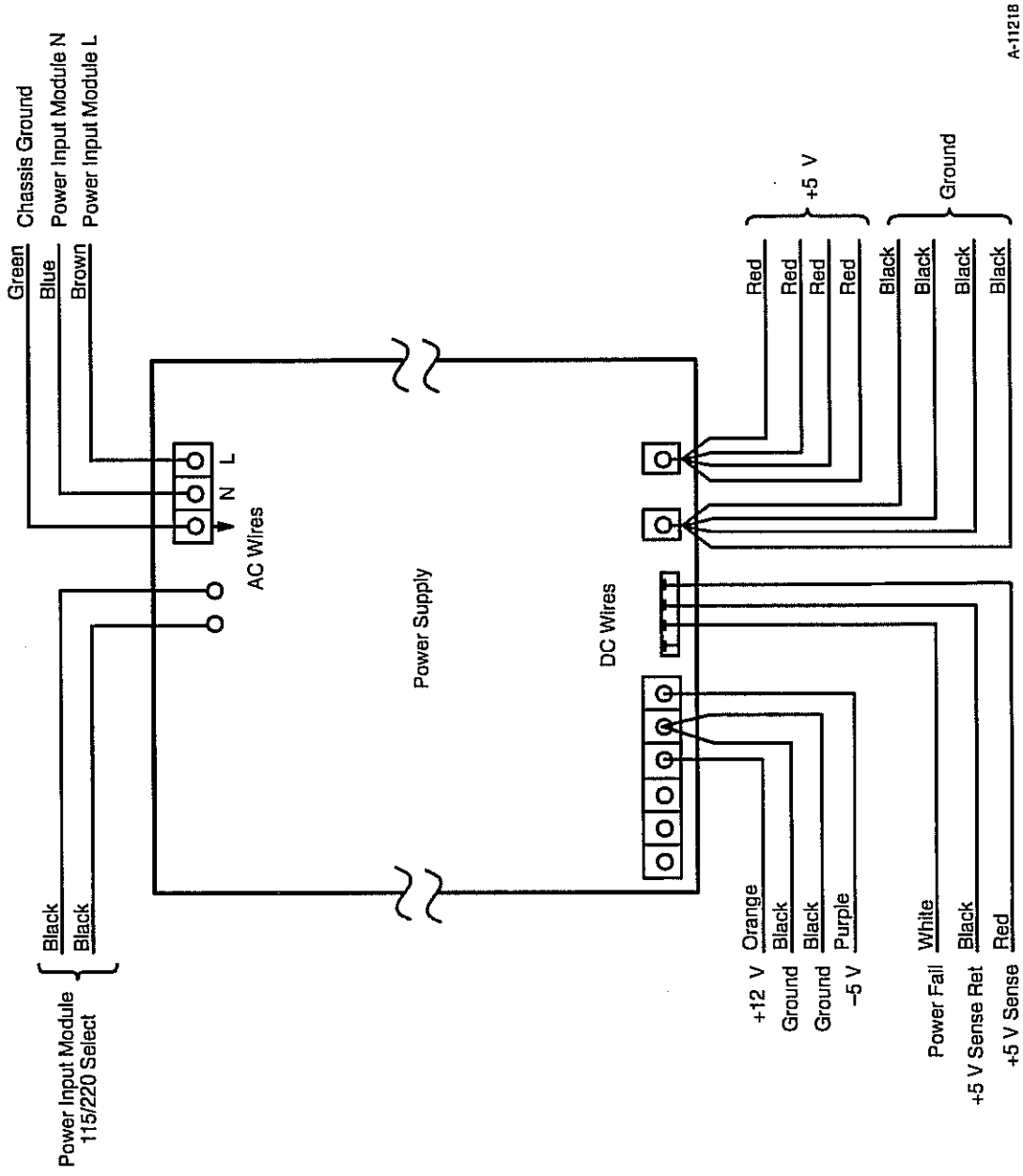


Figure 4-31. Power Supply Wiring Diagram (14-inch Chassis)

PROM and RAM Execution Level

The HK68/V30 controller board is designed to start processor execution from its onboard PROM after a system power-up sequence or after a reset sequence. The functionality available at the PROM execution level is limited and is intended only as the first step in making the CRAY EL series computer system functional. PROM execution enables the following operations: console I/O, maintenance SCSI I/O, software release installation to single-user mode, reading and taking IOS dumps, and IOS booting. Neither the major system peripherals nor UNICOS can be operated at PROM execution level. The execution level that is running can be determined from the prompt level displayed on the system console. If the IOS is under PROM control, the prompt displayed is `BOOT>`. After the load command has been entered, IOS control changes to IOS kernel control. This causes the PROM to boot the kernel from the maintenance hard disk and the console prompt to change to `IOS>`.

The IOS kernel is loaded and executed from the HK68/V30 onboard 4 Mbytes of RAM. The RAM is loaded by executing the `load` command from PROM. This command defaults to the /ios/ios file, which contains the IOS kernel. The processor branches to and executes from the IOS kernel after it is loaded. Executing the IOS kernel sequence initializes the VME hardware in IOS 0, reports the peripheral configuration it locates, then initializes the `load` command in the next IOS via IOSNET. This process continues until all IOSs in the CRAY EL series system have been initialized. After all IOSs have been initialized, the system is completely functional, and the MIOP returns the `IOS>` prompt to the system console and waits for input. This entire process can be initiated by a power-on sequence, by pressing the VME reset button on the control panel, or by issuing a `reload` command from the system console keyboard. For a full 16-IOS system, the initialization process can take up to 15 minutes.

Note that the initialization process polls each VME slave to determine which slaves are connected. If the slave can pass a basic power-on diagnostic test, it acts as a data pass-through for the VMEbus. Consequently, the IOP may not report peripherals because it does not detect them. However, if there are no error signals from the slave, no error is reported to the boot procedure. It is helpful to compare the peripheral report on the system console with the actual system configuration.

In case of an IOS error or failure, failure information that is captured is entered into a file on the IOS disk called `/adm/syslog` that accepts IOS panic messages, IOS warning messages, and IOS notices. The IOS panics are fatal conditions that require the system to be rebooted, and the warnings and notices are nonfatal conditions used to keep a record of

conditions that may require attention. The system log can be accessed by using the `cat` command under PROM execution, or by using the `more` *`filename`* command under kernel execution.